

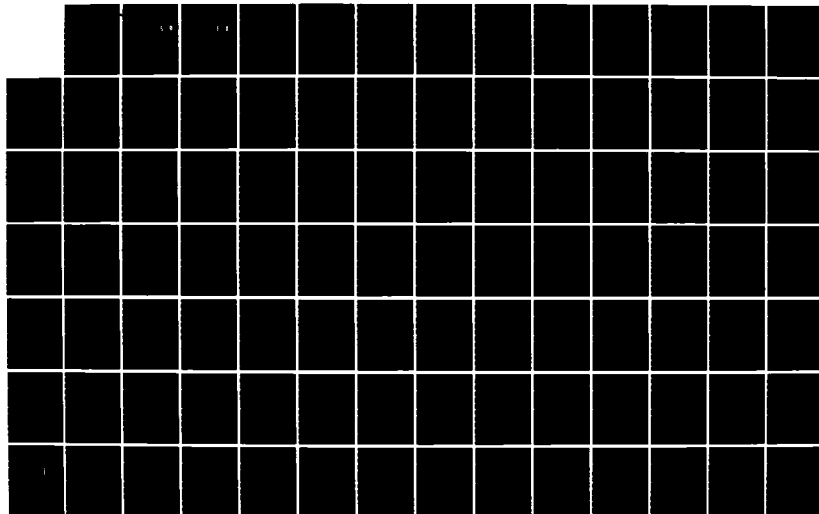
AD-A163 838

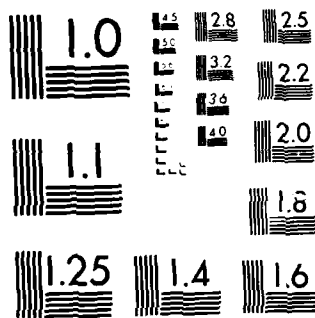
CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE
ESTIMATION FOR TRACKING (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI... A S NETZER
DEC 85 AFIT/GRE/ENG/85D-2 F/G 17/5

1/4

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A163 830

ATTC FILE COPY



DTIC
ELECTE
FEB 10 1986
S D

CHARACTERISTICS OF BAYESIAN MULTIPLE
MODEL ADAPTIVE ESTIMATION FOR TRACKING
AIRBORNE TARGETS

THESIS

Allan S. Netzer
Captain USAF

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

86 2 10 022

AFIT/GAE/ENG/85D-2

DTIC
ELECTE
FEB 10 1986
S D D

CHARACTERISTICS OF BAYESIAN MULTIPLE
MODEL ADAPTIVE ESTIMATION FOR TRACKING

AIRBORNE TARGETS

THESIS

Allan S. Netzer
Captain USAF

AFIT/GAE/ENG/85D-2

Approved for public release; distribution unlimited

AFIT/GAE/ENG/85D-2

CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE
ESTIMATION FOR TRACKING AIRBORNE TARGETS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Aeronautical Engineering

Allan S. Netzer, BAE
Captain, USAF

December 1985

Accession For	
NTIS	CRA&I
DTIC	TAB
Unannounced	
Justification	
By	
Distribution	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited



Preface

This study is another link in a growing chain of research conducted at the Air Force Institute of Technology, to design a tracking algorithm for use with the Air Force Weapons Laboratory's high energy laser weapon system. As such, my effort extends the work done by Capt. P. Loving with the multiple model adaptive Kalman filter/enhanced correlator tracking algorithm. This study investigates the tracker's characteristics in order to evaluate better the tracker's performance in various tracking scenarios.

A great deal of credit belongs to my predecessors without whom an investigation of this complexity could not have been accomplished. I would like to express my deepest thanks to Dr. Peter Maybeck, my thesis advisor. His guidance, motivation, and above all patience, was invaluable to the completion of this study. I would also like to thank Capt. Steve Rogers for his help in the Fourier domain.

I would like to express a special thanks to my family, who kept telling me "you can do it". Most especially, I want to express my love to my wife for supporting me throughout the entire ordeal, and providing the secretarial skills necessary to complete this effort. Finally I would be truly remiss if I did not thank Daisy for all the times she patiently waited for me while I studied, before we could play.

Table of Contents

	Page
Preface	ii
List of Figures	vi
List of Tables	ix
List of Symbols	x
Abstract	xiv
I. Introduction	1
1.1 Background	1
1.2 Problem	6
1.2.1 The Correlator/Linear Kalman Filter Tracker	7
1.2.2 Extended Kalman Filter Tracker	11
II. Multiple Model Adaptive Filtering	14
2.1 Bayesian Multiple Model Adaptive Filtering	16
2.2 Maximum A Posteriori Adaptive Filtering	19
2.3 Implementation of the Multiple Model Algorithm	20
2.4 Summary	21
III. Truth Model	22
3.1 Introduction	22
3.2 Truth State Model	23
3.3 Simulation Space Model	28
3.3.1 Coordinate Frames	29
3.3.2 Velocity Projection on the FLIR Plane	30
3.4 Measurement Model	33
3.5 Target Image Projection on the FLIR Image Plane	37
3.6 Target Trajectories	39
3.7 Spatially Correlated Background Noise	42
3.7 Summary	43
IV. Tracking Algorithm	44
4.1 Introduction	44

4.2	State Space Model	44
4.3	Measurement Model	49
4.3.1	Filter processing in the Fourier Domain	50
4.3.1.1	Target Image Estimation (Template).	50
4.3.1.2	Pseudo-Measurements by Enhanced Correlation .52	
4.3.2	Kalman Filter Update Equations.	54
4.4	FLIR Sensor Pointing Controller . . .	55
4.4.1	Pointing Controller Implementation	56
4.5	Filter Implementation	58
4.6	Summary	60
V.	Algorithm Test Set-Up	61
5.1	Introduction	61
5.2	Tracker Statistics	61
5.3	Performance Plots Format	64
5.4	Greyscale Plots.	68
5.5	Parameter Values	70
5.5.1	Truth Model Parameters	70
5.5.2	Filter Parameters	72
5.6	Designation Code	72
5.7	Summary	75
VI.	Algorithm Investigation	76
6.1	Introduction	76
6.2	Bias Investigation	76
6.2.1	X - Channel Ramping	77
6.2.2	Y - Channel Ramping	81
6.3	Non - Ideal Controller Simulation . .	82
6.3.1	Investigation	83
6.4	Signal-to-Noise Ratio Sensitivity Analysis	91
6.4.1	Results	91
6.5	Target Range From Sensor/Pixel Size Sensitivity Analysis	94
6.5.1	Investigation	94
6.6	Target Shape Investigation	104
6.6.1	High Aspect Ratio Targets . . .	104
6.6.2	Multiple Dispersion Hotspots. .	112
6.6.3	Target/Decoy Experiment . . .	116
6.7	Summary	124
VII.	Conclusions and Recommendations	125
7.1	Introduction	125
7.2	Conclusions	125
7.2.1	Bias Investigation	125

7.2.2	Non - Ideal Controller126
7.2.3	Signal-to-Noise Ratio Sensitivity128
7.2.4	Target Range from Sensor . .	.129
7.2.5	Pixel Size130
7.2.6	Target Shape Investigation .	.131
7.2.7	Target/Decoy Experiment134
7.3	Recommendations135

List of Figures

Figure	Page
1-1 Linear Kalman Filter/Enhanced Correlator Algorithm	9
1-2 Extended Kalman Filter Algorithm	13
2-1 Multiple Model Filtering Algorithm	15
3-1 Target/Inertial Frame Geometry	31
3-2 Azimuth Geometry	31
3-3 Elevation Geometry	32
3-4 Apparent Target Intensity Pattern on the FLIR Image Plane	34
3-5 Hot Spot Distribution	35
3-6 Image Projection Geometry	37
3-7 Trajectory One	40
3-8 Trajectory Two	41
5-1 Typical Plot - Y Dynamics Before Update	65
5-2 Typical Plot - Y Dynamics After Update	66
5-3 Typical Plot - Filter rms Error vs True rms Error	67
5-4 Greyscale Target Image and Template	69
6-1 Greyscale Target Image, Prior to Template Formulation	86
6-2 Greyscale Target Image and Template, Non-Ideal Controller	87
6-3 Greyscale Target Image and Template, Non-Ideal Controller	90
6-4 Greyscale Target Image and Template $z_0 = 5\text{km}$	99
6-5 Greyscale Target Image and Template, New Target Image $z_0 = 5\text{km}$	101

List of Figures (Continued)

Figure	Page
6-6 Greyscale Target Image and Template, New Target Image $z_0 = 5\text{km}$103
6-7 Greyscale Target Images High AR case 1106
6-8 Greyscale Target Image and Template, High AR case 2, Frame 2107
6-9 Greyscale Target Image and Template, High AR case 2, Frame 15109
6-10 Greyscale Target Image and Template, High AR case 2, Frame 30110
6-11 Greyscale Target Image and Template, High AR case 2, Frame 60111
6-12 Greyscale Target Image and Template, High AR case 4113
6-13 Greyscale Target Image and Template, Target/ Decoy Experiment Frame 2117
6-14 Greyscale Target Image and Template, Target/ Decoy Experiment, Frame 31119
6-15 Greyscale Target Image and Template, Target/ Decoy Experiment, Frame 33120
6-16 Greyscale Target Image and Template, Target/ Decoy Experiment, Frame 36121
6-17 Greyscale Target Image and Template, Target/ Decoy Experiment, Frame 45122
6-18 Greyscale Target Image and Template, Target/ Decoy Experiment, Frame 60123

Performance Plots

Bias Investigation	Appendix A
Non-Ideal Controller Investigation	Appendix B
Signal to Noise Ratio Sensitivity Analysis . .	Appendix C

Performance Plots (Continued)

Target Range From Sensor/Pixel Size

Sensitivity Appendix D

Target Shape Sensitivity Analysis Appendix E

List of Tables

Table	Page
4-1 Filter Tuning	59
6-1 Controller Test Points	84
6-2 S/N Time Averaged Statistics for .5 to 2.0 seconds	92
6-3 S/N Time Averaged Statistics for 3.5 to 5.0 seconds	92
6-4 High Aspect Ratio Target Test Points	104
6-5 Multiple Dispersion Hotspot Time Averaged Statistics for .5 to 2.0 seconds.	114
6-6 Multiple Dispersion Hotspot Time Averaged Statistics for 3.5 to 5.0 seconds	115

List of Symbols

Symbol

AR	aspect ratio
\underline{A}	direction cosine matrix
A_p	area of a single pixel
A,B	atmospheric jitter model break frequencies
\underline{a}	vector of uncertain parameter values
α	smoothing constant in averaging process
$\alpha(t)$	azimuth
$\beta(t)$	elevation
\underline{B}	control input matrix
E	sample mean error
$E(\)$	expected value
\underline{e}	unit vector
e	error between truth model system output and filter system output
\underline{F}	system plant matrix
$F(\)$	Fourier transform operation
$F^{-1}(\)$	inverse Fourier transform operation
f_x, f_y	spatial frequencies
\underline{G}	system noise input matrix
G,H	complex-valued functions in spatial frequency domain
g,h	function in the spatial domain
$\underline{h}(\)$	nonlinear intensity function
$\underline{H}(\)$	linearized intensity function
\underline{H}	system output matrix, $\partial \underline{h} / \partial \underline{x}$

List of Symbols (Continued)

Symbol

I	identity matrix
I_{\max}	maximum intensity of a hot-spot
K	Kalman filter gain
\underline{I}	system state transition matrix
$N \times N$	size of data array to be processed
p	range to target
\underline{P}	state covariance matrix or dispersion matrix of target image
p	conditional probability
\underline{Q}_F	filter dynamics driving noise intensity matrix (dynamics and atmospheric)
\underline{Q}_{DF}	filter dynamics driving noise intensity matrix (on noises driving acceleration, velocity, and position state differential equations)
\underline{Q}_{Ad}	discrete noise covariance kernel descriptor for truth model atmospheric states
\underline{R}	measurement noise covariance matrix
\underline{r}	residual vector
r_h	horizontal range
σ	standard deviation of a process or dispersion of an intensity distribution
τ_{AF}	correlation time assumed for atmospheric jitter
τ_{DF}	correlation time assumed for target dynamics for the filter
t	time
Δt	sample time
θ	orientation angle in the FLIR image plane

List of Symbols (Continued)

Symbol

\underline{u}	deterministic control input
\underline{v}	measurement noise vector - zero-mean, white Gaussian noise
\underline{v}_I	inertial velocity vector
\underline{v}'	unit variance, zero mean, white, Gaussian noise vector
\underline{w}	white Gaussian noise vector driving the dynamics model
\underline{x}	general state vector
$\underline{x}_I, \underline{y}_I, \underline{z}_I$	inertial axes
δ	angle between inertial velocity vector and the plane that is perpendicular to the line of sight
(x, y)	FLIR frame coordinates
$y(t)$	current averaged data frame
$y(t)$	current data frame
\underline{Z}	measurement history
\underline{z}	measurement vector
\perp	perpendicular
LOS	line of sight
rms	root mean squared
c	Cholesky square root
r	error due to non-ideal pointing
r(a)	fraction of the commanded input that is realized due to non-ideal pointing

Subscripts

A	atmospheric disturbances
AF	filter model for atmospheric jitter

List of Symbols (Continued)

Subscripts

α	azimuth direction
β	elevation direction
C	centroid
D	target dynamics
DF	filter model for target dynamics
d	discrete time
F, f	filter
I	inertial reference frame
i	i-th time frame
k	k-th term or filter
kl	kl-th pixel
m	m-th hot-spot or ellipsoid
o	initial value
pv	direction perpendicular to the target velocity
ppv	direction mutually perpendicular to the target velocity and pv direction above
r	direction along line of sight to the target from the tracker
T	truth model
v	direction of the target velocity vector

Superscripts

c	value after control application
$\hat{}$	estimate
-	before measurement update
+	after measurement update

Abstract

Previous studies at the Air Force Institute of Technology have led to the development of a multiple model adaptive filter (MMAF) tracking algorithm which provides significant improvements in tracker performance against highly-dynamic airborne targets, over the currently used correlation trackers. A forward looking infra-red (FLIR) sensor is used to provide a target shape function to the tracking algorithm in the form of an 8 x 8 array of intensities projected onto a field of view (FOV). This target image measurement is correlated with an estimate of the target image, a template, to produce linear offset pseudo-measurements from the center of the FOV, which are provided as measurements to a bank of linear Kalman filters, in the multiple model adaptive filtering (MMAF) structure. The output of the MMAF provides the state estimates used in pointing the FLIR sensor, and generating the new target image estimate. This study investigates the characteristics of this algorithm in order to evaluate its performance against various target scenarios.

I. INTRODUCTION

Since the conception of lasers in the late 1950's, the idea of laser weapons has been generally considered science fiction by the public at large. With the recent advances in laser technology, the laser has been highly successful in many military applications as well as medical and industrial. The ability to transmit energy almost instantaneously onto a target makes it an attractive potential weapon system.

With the inception of the Strategic Defense Initiative (SDI), the laser has been identified as a potential weapon system warranting further investigation. With current laser inefficiencies, limited energy is available in the beam. This requires tight specifications for laser pointing and tracking systems, in order to deposit sufficient energy on a point to achieve damage to the target. This requirement has motivated research into innovative methods of accurately tracking highly maneuvering targets, at high velocities.

1.1 BACKGROUND

The Air Force Weapons Laboratory at Kirtland AFB, New Mexico, is currently developing high energy laser weapons to be used against airborne targets and other vehicles. Target measurements are obtained by means of a forward looking

infra-red sensor (FLIR). These measurements are used to track the target passively, thereby preventing the target from detecting that it is being tracked.

The target measurement provides information about target motion. However, these measurements are corrupted by several sources. These include: atmospheric jitter, sensor measurement errors, background clutter, and mirror vibration in the laser pointing.

Currently, pointing and tracking tasks are accomplished by means of a correlation algorithm. This algorithm compares FLIR measurement data from the previous sample time to the current data. Cross correlation of the data establishes relative position offsets. The offsets are assumed to be due to the target motion and the FLIR is pointed to center the target in its field of view (FOV).

Although the correlation tracker performs reasonably well against a wide variety of targets, it has several inherent limitations. The algorithm has no provisions to distinguish between actual target motion and apparent target motion due to signal corruption. Additionally, changing target shapes due to changing of the FLIR/Target orientation can be interpreted as target motion. Another limitation of correlation trackers is time lag due to a finite time requirement for cross correlation and pointing. The algorithm provides no estimation of future positions. These limitations motivate investigation of alternate tracker

algorithms.

Since 1978, the Air Force Institute of Technology has supported a number of papers and Master's theses demonstrating the feasibility and performance benefits of tracking algorithms based on Kalman filtering techniques. Kalman filter characteristics directly address the correlator limitations previously discussed. Using the statistical characteristics of atmospheric jitter and measurement errors, and a model of the anticipated target dynamics, the filter estimates the target position at the next sample time, from the previous history of measurements. This estimate accounts for apparent target motion due to atmospherics as previously discussed, in essence filtering the noise corrupted measurement. The prediction allows the FLIR to anticipate target motion, thus reducing tracking error due to time delays or pointing system dynamical lags.

The initial feasibility study by Mercier [9,12] demonstrated performance benefits of an extended Kalman filter algorithm against long range targets. The long range point source was assumed to be well modeled as a bivariate Gaussian distribution. The 4-state filter used a first order zero-mean Gauss-Markov position model to portray benign target dynamics as seen in the FLIR image plane. The filter measurement noise due to internal FLIR errors and background clutter was modeled as uncorrelated in time and space. This algorithm increased tracking performance an

order of magnitude over the correlation algorithm, in benign scenarios.

Work by Harnly and Jensen [2,7,8] incorporated estimates of target velocities and accelerations to enable tracking of more maneuverable targets. Target image equal-intensity profiles were modeled as being elliptical rather than circular as in Mercier's research, and adaptive estimation of the target shape was incorporated.

Research by Singletary [15] and Rogers [10,14] implemented algorithms which had no prior knowledge of, or assumptions about target shape. The filter was tested against multiple hot-spot targets with dynamic variations.

Rogers also developed an alternative filter algorithm which used an enhanced correlator to obtain offsets from the FLIR measurements relative to a template. The template was composed by averaging centered target images from previous measurements to estimate the target shape. The centering process was accomplished via filter estimates of the target location within the FLIR field-of-view. These offsets were then used as measurements to a linear Kalman filter. The reduced computational loading and comparable performance of the linear filter/correlator [10,14] make it preferable over the extended Kalman filter for many scenarios.

Follow-on research by Millner [13] and Kozemchak [3], tested both the extended Kalman filter and the linear filter/correlator algorithm against close range, highly

maneuvering targets. Both filters were slow to respond to harsh maneuvers significantly different from the filter's target dynamics model, yielding difficulty in maintaining lock on targets performing maneuvers in excess of 5 g's.

In order to improve upon this limitation, Flynn [1] initially investigated a multiple model adaptive filter (MMAF). The MMAF was later successfully implemented by Suizu [16]. The filter contained a bank of 2 filters, one tuned for highly maneuvering targets, the other for benign targets, to change the targets dynamics model adaptively. The filter based on a highly maneuvering model included a larger FOV, to aid the filter in maintaining lock of highly maneuvering targets. Using probabilistic weighting, the filter adaptively changed target dynamics model and FOV size, increasing the filter performance to allow tracking of targets pulling 20 g's at 20 km. Both the extended Kalman filter and the linear filter/correlator were tested as the form of "elemental" filter within the MMAF bank, with similar results.

Follow-on research by Loving [4] added an additional filter to the MMAF bank, based on intermediate levels of target dynamics to aid in the tracking of highly maneuvering targets. Additionally, a Maximum a Posteriori (MAP) algorithm was developed as well as the Bayesian approach previously used. The MAP algorithm utilized the same MMAF bank as the Bayesian filter, however, it produced an

estimate from the one elemental filter with the highest probability of validity, rather than forming a probabilistically weighted average of all elemental filter estimates. The addition of the third elemental filter in the MMAF Bank showed significant improvement in tracking performance. Both the Bayesian and MAP estimation techniques supported accurate tracking of highly dynamic airborne targets, with little significant performance variations realized between the two.

1.2 PROBLEM

This effort concentrates on expanding the results obtained by Loving [4], using the linear filter/correlator algorithm developed by Rogers [10,14]. The potential for decreased computational loading compared to the extended Kalman filter, while maintaining comparable accuracy, makes this filter algorithm more attractive for further development.

Several significant biases and apparently divergent trends in filter error were observed in previous work. Investigation of these results to identify modeling errors or filter inadequacies are performed, in order to improve filter performance.

In all previous work, the repositioning of the FLIR sensor was assumed to be accomplished perfectly in less than

one sample period. In this work, effects of more realistic feedback controllers will be investigated. This will be accomplished by incorporating time lags, modeling inertial and servo effects, in the pointing of the FLIR sensor.

A sensitivity analysis is conducted. A major aim of the studies is to determine tracker characteristics in order to provide insights into enhancing filter performance. In addition, performance capabilities not previously investigated are studied to establish fundamental limits of performance, and to address the issue of being able to meet strict performance specifications. Robustness studies (filter not knowledgeable of parameter variations), is another objective of the analysis. Parameters to be studied will include: signal to noise ratio, range to target, and pixel size sensitivity.

An investigation of tracking performance for various target shapes will be conducted. The objective is to identify tracker characteristics, and performance, with respect to various image functions. This will provide insights into image configurations that provide the least and most difficulty to the tracker, as well as to discern possible means of defeating the tracker.

1.2.1 THE CORRELATOR/LINEAR KALMAN FILTER TRACKER. The correlator/linear Kalman filter developed by Rogers [10,14] uses pseudo-measurements obtained by processing FLIR data

with an enhanced correlator to update the state estimate. The state estimate is then propagated forward in time based on the filter's estimate of the target dynamics. This estimate of the future target position is then used to accomplish the pointing and control task for the FLIR/laser. Figure 1-1 shows the algorithm structure for a single filter, which could represent one of the elemental filters in the MMAF bank, described in the next chapter.

The tracking algorithm uses an 8 x 8 array of target intensities obtained by the FLIR measurement, to establish a 64 element shape function from the target intensity profile. This shape function is correlated against a template made up of previous shape functions that have been centered on the FLIR image plane. The x and y offsets which are obtained by correlating the shape function and the template are input to the Kalman filter as linear pseudo-measurements. The measurements are used along with the filter's target dynamics model to estimate the state at the next sample period $\hat{x}(t_{i+1}^-)$. It is then desired to center the FLIR field of view (FOV) at this point in preparation for the next measurement.

The template generation portion of the algorithm begins with the raw data obtained from the FLIR sensor. This data is Fourier transformed to allow for the comparative ease of performing the necessary computations in the frequency domain, and to allow for optical processing eventually.

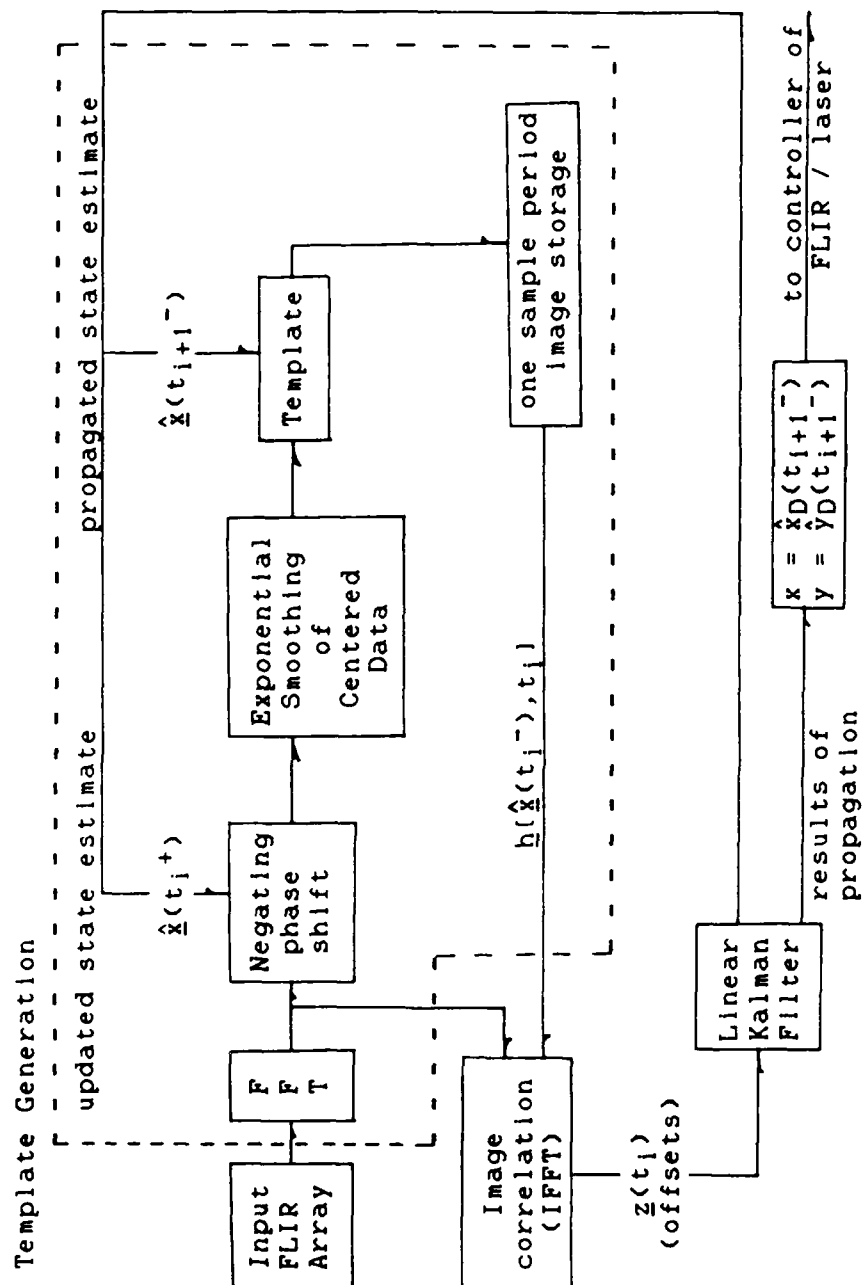


Figure 1-1. Linear Kalman Filter/Enhanced Correlator Algorithm

The 8 x 8 pixel array is expanded to a 24 x 24 pixel array. This array contains the original 8 x 8 array centered with additional FLIR data (rather than zeros, as is often used) padded along the outer frame. The 24 x 24 pixel data array reduces the effects of edges, aliasing, and leakage conditions encountered in transforming a finite sequence via FFT techniques. The data is then shifted to align the filter's estimate of the target centroid with the center of the current FOV. The centered data is then temporally averaged with the previous frames of transformed and centered data; this is accomplished by means of exponential smoothing rather than finite memory averaging so as not to require storage and processing of many frames of data. Inverse fast Fourier transforming (IFFT) this generates an estimate of the shape function, the template, in the spatial domain. Actual filter implementation maintains the template in the Fourier domain for correlation (IFFT) with the current Fourier transformed FLIR intensity shape function.

The correlation of the template $\langle \hat{x}(t_i^-), t_i \rangle$ shown in figure 1-1) and the current intensity function provides the measurements for the linear Kalman filter to use for updating the state estimates, as discussed in Chapter 4. The filter states include the target position estimates (as seen in the FLIR image plane) due to true target dynamics as well as position error, due to atmospheric distortion of IR phase fronts. The FLIR

pointing controller positions the center of the FOV at the propagated filter estimate of the true vehicle position one sample period into the future. On the other hand, the position estimate of the data on the FLIR image plane is determined by including the error effects due to atmospheric turbulence. Thus, if there were only a single filter, the expected location of the intensity profile relative to the center of the FOV is the position offset predicted by the atmospheric turbulence states estimates. The modifications required to replace a single filter with the multiple model adaptive filtering algorithm are discussed in Chapter 4. The filter measurements are obtained by correlation of the FLIR data with the template, and the Kalman filter update is then incorporated into the state estimate establishing $\hat{x}(t_i^+)$. This estimate is used then to center the FLIR intensity profile to be included in the template for the next sample.

1.2.2 EXTENDED KALMAN FILTER TRACKER. The extended Kalman filter algorithm was the first tracker in the investigation of Kalman filter trackers. It utilizes the average intensity over each of the 64 pixels in the 8 x 8 FLIR array as a filter measurement. Due to the filter measurement arrangement, the intensity function $h[\hat{x}(t_i^-), t_i]$ is required, establishing the need for the extended Kalman filter to incorporate the nonlinear measurement update. The data processing algorithm is presented in Figure 1-2.

Except for the need to calculate the linear and nonlinear intensity functions $H(t_i)$ and $h[\hat{x}(t_i^-), t_i]$ the algorithm is very similar to the previous one. Due to the attractive characteristics of the linear Kalman filter/correlator, the extended Kalman filter is not developed in this effort. Filter development can be found in [2,7,8,10,14,16].

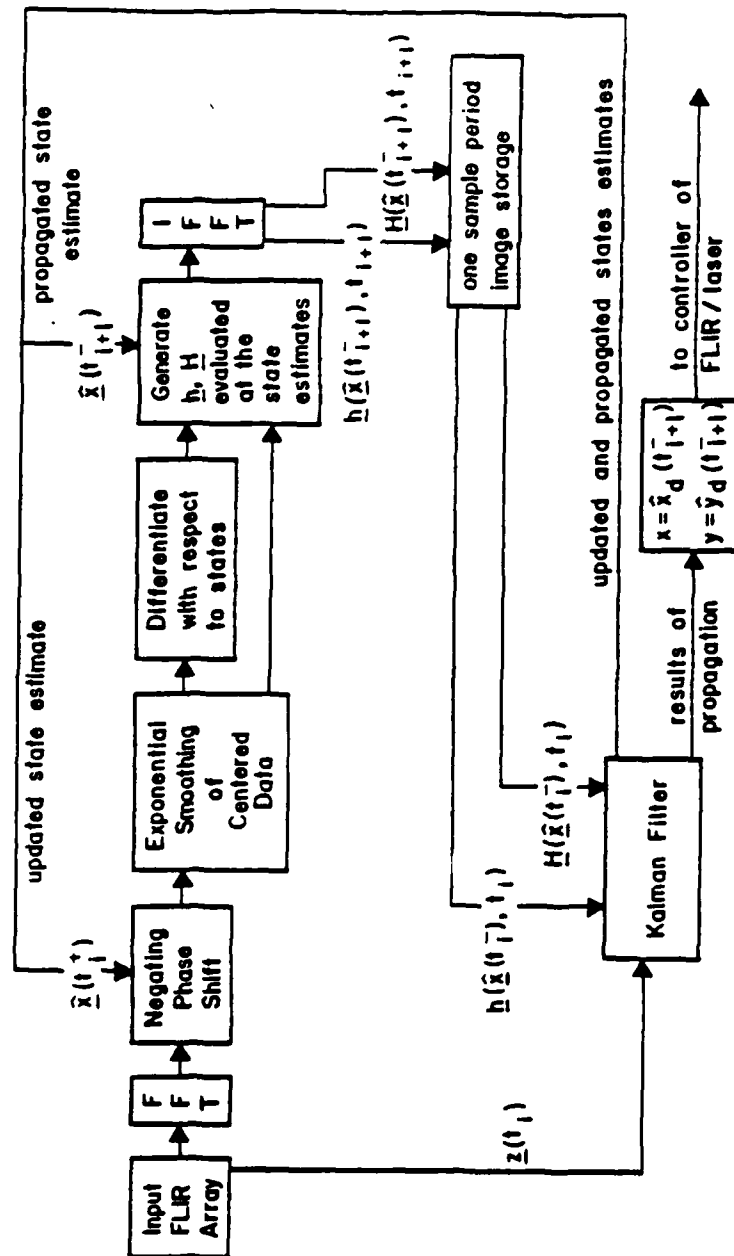


Figure 1-2. Extended Kalman Filter Algorithm

II. MULTIPLE MODEL ADAPTIVE FILTERING

To achieve a high level of performance in a single Kalman filter tracker, it would be necessary to match the uncertain parameters of the dynamics model to the dynamics of the target. Since the range of parameters which provides optimal performance is continuous, it is necessary to discretize the parameter space to keep the algorithm tractable. For a target which displays n significantly different discrete sets of characteristic dynamics, no one vector of parameters, \underline{a} , is adequate. It is then desirable to match the k^{th} possible parameter vector value \underline{a}_k , to the k^{th} target dynamic characteristic, to achieve maximum performance. The multiple model adaptive filter (MMAF) consists of a bank of n independent Kalman filters processed in parallel. Each filter is "tuned" for a discrete characteristic target dynamics by the appropriate \underline{a}_k . At each sample time, the residuals of all filters are used to calculate conditional probabilities identifying which filter has the highest probability of the best performance: the probability that \underline{a} assumes the value of \underline{a}_k , conditioned on the observed measurement history. This conditional probability is called the hypothesis conditional probability $p_k(t_1)$. The MMAF structure is presented in Figure 2-1.

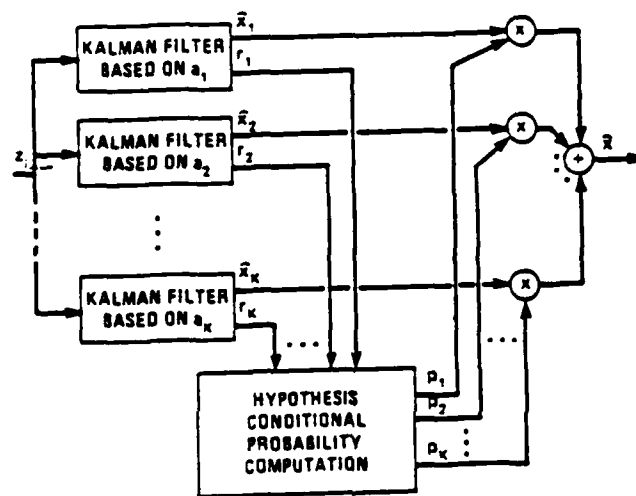


Figure 2-1. Multiple Model Filtering Algorithm

This chapter heuristically presents the MMAF techniques implemented in this study. Rigorous mathematical developments are presented in references [4,6,16].

2.1 BAYESIAN MULTIPLE MODEL ADAPTIVE FILTERING

The Bayesian MMAF estimation consists of probabilistic weighting of all n filters using the hypothesis conditional probability $p_k(t_i)$. This probability is determined recursively at each sample period, for each of the n filters. The recursion is developed in [4,6,10,14,16], for the k th filter as:

$$p_k(t_i) = \frac{n \cdot f_{\underline{z}(t_i) | \underline{a}, \underline{Z}(t_{i-1})}(\underline{z}_i | \underline{a}_k, \underline{Z}_{i-1}) \cdot p_k(t_{i-1})}{\sum_{j=1}^n f_{\underline{z}(t_i) | \underline{a}, \underline{Z}(t_{i-1})}(\underline{z}_i | \underline{a}_j, \underline{Z}_{i-1}) \cdot p_j(t_{i-1})} \quad (2-1)$$

where

$$f_{\underline{z}(t_i) | \underline{a}, \underline{Z}(t_{i-1})}(\underline{z}_i | \underline{a}_k, \underline{Z}_{i-1}) = \frac{\exp(\cdot)}{(2\pi)^{m/2} |\underline{A}_k(t_i)|^{1/2}} \quad (2-2)$$

$$(\cdot) = -1/2 \underline{r}_k^T(t_i) \underline{A}_k^{-1}(t_i) \underline{r}_k(t_i) \quad (2-3)$$

$$\underline{A}_k = \underline{H}_k(t_i) \underline{P}_k(t_{i-1}) \underline{H}_k^T(t_i) + \underline{R}_k(t_i) \quad (2-4)$$

$$\hat{\underline{x}}(t_i^+) = \sum_{k=1}^n \hat{\underline{x}}_k(t_i^+) \cdot p_k(t_i) \quad (2-5)$$

and

\underline{a}_k = the parameter value assumed in the k th filter

$\underline{r}_k(t_i)$ = the k th filter residual, $[\underline{z}(t_i) - \underline{H}_k(t_i) \hat{\underline{x}}_k(t_{i-1})]$

As can be seen from Equation (2-1), $p_k(t_i)$ is the ratio of two products. The numerator is the k^{th} filter's product of its previous hypothesis probability and the conditional probability density of the current measurement given the k^{th} filter's assumed parameter value and the previous measurement history. The denominator is the sum of the same products for all n filters. When the k^{th} filter is the best match for the current target dynamics, that filter will produce the smallest squared residual relative to the filter-computed residual covariance of the n filters, i.e. the smallest quadratic in Equation (2-3). The smaller residuals will cause Equation (2-3) to become a smaller negative quantity, causing Equation (2-2) to be larger for the k^{th} filter than for the other $n-1$ filters. The ratio then formed by Equation (2-1) will be the largest for the k^{th} filter, causing its probability to converge to the largest value.

As can be seen in Figure 2-1, each of the n filters processes its own estimates and residuals in parallel. The recursion is then run at each sample time and a $p_k(t_i)$ for each filter assigned. Equation (2-5) is used to determine the MMAF weighted state estimate.

As previously stated, the n filters are each based on a model representing a discrete dynamic uncertainty significantly different from the other models. It is assumed that the filter which represents the closest to the

true target dynamics will produce significantly smaller residuals relative to the filter-computed covariance than the mismatched filters. By this mechanism, Equation (2-1) will produce the heaviest weightings for the best filter. In order for a significant difference in residuals to be realized, each filter must be specifically tuned for best performance against a discrete target trajectory that matches its internal dynamics model. The common practice of adding pseudo-noise to compensate for linear model inadequacies in single Kalman filter applications should be avoided, since it tends to blur the distinctions between the estimates (and residuals) based on different models.

In addition, the calculated probabilities should have artificial lower bounds enforced [6]. This is to prevent the mismatched filter's $p_k(t_i)$ values from converging to zero. Once a filter p_k is allowed to reach zero, it will remain zero for all time; likewise reaching very small values results in great difficulty increasing that p_k via Equation (2-1). This effectively removes that filter from the bank. The loss of a filter could significantly affect the MMAF future performance in the event that the target's future dynamics would best match that filter's dynamics model. A lower bound of .001 was established by Loving [4] for this application and will be continued in this effort. It is noted that a larger p_k lower bound allows for the faster transition to a filter with residuals that indicate

that a heavier p_k is appropriate. However, this faster transition is at the expense of an inappropriately higher weight on "incorrect" filters in steady state, reducing MMAF performance.

2.2 MAXIMUM A POSTERIORI ADAPTIVE FILTERING

The maximum a posteriori (MAP) multiple model state estimator was implemented by Loving [4]. The MAP filter consists of a bank of independent Kalman filters as in the Bayesian filter. The residuals from all filters running in parallel are used to calculate $p_k(t_i)$ as shown in Section 2.1. However, unlike the Bayesian MMAF state estimate of Equation (2-5), the MAP adaptive estimate is taken from the one elemental filter with the highest $p_k(t_i)$. This best filter is used on a sample by sample basis until another filter is identified as having the highest $p_k(t_i)$. This is as opposed to the Bayesian MMAF, which is the optimally weighted average of all elemental filters, i.e. the conditional mean rather than the conditional mode. The MAP filter was investigated with expectations of faster response to a changing target dynamics. Since no influence from mismatched filters are included in the MAP estimate, (provided that the $p_k(t_i)$ computation selects the "right" filter), higher performance against "design point" trajectories was expected. On the negative side, the MAP

was expected to realize reduced performance against target trajectories between design points since it does not blend elemental filter results together. Since computational loading limits the number of discrete dynamics uncertainties, no one filter in the bank may be a good match to the true target behavior, and coarse discretization would be expected to degrade performance of the MAP filter more than the Bayesian form of the filter. Results obtained by Loving [4] showed that no significant performance advantage was achieved by either approach.

2.3 IMPLEMENTATION OF THE MULTIPLE MODEL ALGORITHM

The MMAF for this effort consists of a bank of three elemental filters. By varying the filter's dynamics correlation time, τ_{DF} , and the dynamics driving noise strength, Q_{DF} , used to model the target acceleration, the elemental filters are independently tuned for three tracking scenarios. τ_{DF} and Q_{DF} are defined precisely in Chapter 4.

The first filter is tuned for a benign target trajectory with the "small" FOV. Each of the 8 x 8 pixels in the FOV is 20 by 20 micro-radians. The second filter is tuned for a highly maneuvering target, pulling ± 20 g's. The assumed trajectories for tuning are defined in detail in Chapter 3. The second filter uses the "larger" FOV in which each pixel in the 8 x 8 array is 60 micro-radians on each

side. The large field of view allows the tracker to maintain lock while tracking high g-maneuvering, where the larger errors would have caused the target image to be off the small FOV. The third filter is tuned for a 10 g-maneuvering target. It uses the small field of view as with the first elemental filter, since experience shows that tracking errors for this scenario are small enough to maintain the image on the small FOV. The filter algorithm and tuning is further discussed in Chapter 4.

2.4 SUMMARY

This chapter has presented a heuristic discussion of multiple model adaptive filtering (MMAF). The purpose of this discussion is to present the motivation for MMAF in this application, as well as the recursion used in the filter implementation.

III. TRUTH MODEL

3.1 INTRODUCTION

The truth model is the simulation of the "real world". It provides the standard against which the filter's performance can be evaluated. For this reason it must model the actual processes of interest as closely as possible. The processes of interest in this study are atmospheric jitter, target dynamics, target shape effects, and background and internal FLIR noises. These processes are important as they affect the filter's perception of target motion. Sensor vibrational effects can be important as additional noise in the FLIR measurement model. This effect however is not considered in this study, since a ground based tracker is assumed.

The FLIR sensor measurement provides average intensity values seen in individual picture elements (pixels), indicating apparent target position at a given time. The term apparent is used here to denote the corrupting effects of the atmospheric distortion. As the radiation from the target passes through the atmosphere, its phase front is distorted, providing a translational shift in the apparent target centroid seen in the FLIR image plane. For a target whose centroid position has changed the amount x_D , due to dynamics, the apparent motion of the centroid as seen by the

FLIR sensor is

$$x_C = x_D + x_A \quad (3-1)$$

where

x_C = x-coordinate of target centroid observed by FLIR sensor;

x_D = x-coordinate of the change in position due to target dynamics;

x_A = x-coordinate of the apparent change in centroid due to atmospheric;

and similarly for y. This apparent target position is measured in units of pixels on the FLIR image plane.

The truth model propagates the dynamics and atmospheric states to define the true target states as well as true apparent position. This chapter discusses the measurement and target models which make up the truth model for the simulation. This discussion includes: the target state space model, the various coordinate frames, multiple hotspot target intensity functions, inertial target trajectories, and noise effects.

3.2 TRUTH STATE MODEL

The target motion in the truth model is described by the linear stochastic differential equation [5]:

$$\dot{\underline{x}}(t) = \underline{F} \underline{x}(t) + \underline{B} \underline{u}(t) + \underline{G} \underline{w}(t) \quad (3-2)$$

where $\underline{x}(t)$ is the state vector made up of target position and atmospheric states. The $\underline{B} \underline{u}(t)$ term is a deterministic input which consists of velocity components to direct the target along the specified trajectory. The $\underline{G} \underline{w}(t)$ term contains the white noise uncertainty, which is applied to the atmospheric state equations.

The solution to the governing Equation (3-2), for a sample data system is:

$$\underline{x}(t_{i+1}) = \underline{\Phi}(t_{i+1}, t_i) \underline{x}(t_i) + \underline{B}_d \underline{u}_d(t_i) + \underline{G}_d \underline{w}_d(t_i) \quad (3-3)$$

where the subscript d denotes the discrete time representation of the appropriate term [5] and $\underline{\Phi}(t_{i+1}, t_i)$ is the state transition matrix associated with \underline{F} in Equation (3-2). The discrete model of the input, $\underline{u}_d(t_i)$, in Equation (3-3) is held constant over a sample period. Additional equivalent discrete versions of terms in Equation (3-2) are defined as:

$$\underline{B}_d = \int_{t_i}^{t_{i+1}} \underline{\Phi}(t_{i+1}, \tau) \underline{B}(\tau) d\tau$$

and \underline{w}_d has covariance

$$\underline{Q}_d(t_i) = \int_{t_i}^{t_{i+1}} \underline{E}(t_{i+1}, \tau) \underline{G}(\tau) \underline{Q}(\tau) \underline{G}(\tau)^T \underline{E}(t_{i+1}, \tau) d\tau$$

where $\underline{Q}(t)$ is the strength of $\underline{w}(t)$.

The state vector is made up of two positions, x_T and y_T and six atmospheric states, in the FLIR plane. Consider a spherical reference frame with the FLIR sensor at the origin. At any time the FLIR FOV is assumed to be a plane tangent to a sphere with radius equal to the sensor range to the target. This $\alpha - \beta$ plane is perpendicular to the line of sight for all time. Assuming the target, i.e. $\alpha - \beta$ plane, is far away from the FLIR sensor, the FLIR azimuth angle, α , and elevation angle, β , can be considered the linear translational coordinates x and y of the target centroid in the FLIR ($\alpha - \beta$) plane. This convention allows for truth model states independent of FLIR pointing.

The deterministic portion of the dynamics consists of the velocities input in the $\underline{B} \underline{u}(t)$ term. The velocities are input relative to the FLIR plane and are of the form for the sampled data system

$$\underline{B} \underline{u}(t_i) = \underline{I} \Delta t [\dot{\alpha}(t_i) \dot{\beta}(t_i)]^T \quad (3-4)$$

The deterministic portion of the state propagation is then:

$$\begin{bmatrix} x(t_{i+1}) \\ y(t_{i+1}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x(t_i) \\ y(t_i) \end{bmatrix} + \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} \Delta t \quad (3-5)$$

This particular form of generating deterministic truth model target trajectories was chosen to be consistent with standard state space modeling, and to allow the option of additional driving noise to produce stochastic process truth model trajectories instead of purely deterministic ones.

The atmospherics developed by Mercier [9,12], are modeled as a third order Gauss Markov process in both the x and y FLIR directions. In the x-direction, the shaping filter is:



where

w_A = unit strength white Gaussian noise

K = system gain, adjusted for desired atmospheric RMS value

A = break frequency; 14.14 rads/sec

B = break frequency; 659.5 rads/sec

x_A = output of shaping filter

The y direction shaping filter is identical. The governing atmospheric stochastic differential equation can be written as

$$\dot{\bar{x}}_A(t) = F_A \bar{x}_A(t) + G_A w_A(t) \quad (3-6)$$

where

$\underline{x}_A(t)$ = the atmospheric state vector

\underline{F}_A = atmospheric system plant matrix

\underline{G}_A = atmospheric noise input matrix

$\underline{w}_A(t)$ = vector of white Gaussian noise inputs with statistics

$$E(\underline{w}_A(t)) = \underline{0}$$

$$E(\underline{w}_A(t) \underline{w}_A^T(t + \tau)) = \underline{Q}_A(t) \delta(\tau)$$

the solution of which is, for a sample data system:

$$\underline{x}_A(t_{i+1}) = \underline{\Phi}_A(t_{i+1}, t_i) \underline{x}_A(t_i) + \underline{G}_{Ad} \underline{w}_{Ad}(t_i) \quad (3-7)$$

where

$\underline{\Phi}_A(t_{i+1}, t_i)$ = atmospheric state transition matrix associated with \underline{F}_A

$$\underline{G}_{Ad} = \underline{C} \sqrt{\underline{Q}_{Ad}}$$

The discrete time white Gaussian noise $\underline{w}_{Ad}(t_i)$, is of the form

$$E(\underline{w}_{Ad}(t_i)) = \underline{0}$$

$$E(\underline{w}_{Ad}(t_i) \underline{w}_{Ad}^T(t_j)) = \underline{I} \delta_{ij}$$

$$\text{and } \underline{C} \sqrt{\underline{Q}_{Ad}(t_i)} \cdot \underline{C} \sqrt{\underline{Q}_{Ad}(t_i)}^T = \underline{Q}_{Ad}(t_i)$$

where $\underline{C} \sqrt{\underline{Q}_{Ad}(t_i)}$ represents the Cholesky square root of $\underline{Q}_{Ad}(t_i)$ [5]

Here again the d subscript denotes the discrete equivalent of a continuous time vector process [5].

The total system is then formed by augmenting the target dynamics states and atmospheric states. The target and atmospheric augmented state propagation is then of the form:

$$\begin{aligned} \underline{x}(t_{i+1}) = & \underline{\Xi}(t_{i+1}, t_i) \underline{x}(t_i) + \begin{bmatrix} \underline{B}_d(t_i) \\ 0 \end{bmatrix} \underline{u}_d(t_i) \\ & + \begin{bmatrix} 0 \\ \frac{1}{\sqrt{Q_{Ad}}} \end{bmatrix} \underline{w}_{Ad}(t_i) \end{aligned} \quad (3-8)$$

where

$\underline{x}(t_i)$ = augmented state vector (two dynamic states and six atmospheric)

$\underline{\Xi}(t_{i+1}, t_i)$ = augmented 8x8 state transition matrix

The state transition matrix as developed by Harnly & Jensen [2,7,8] is

$$\underline{\Xi}(t_{i+1}, t_i) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & e^{-A\Delta t} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{-B\Delta t} & \Delta t e^{-B\Delta t} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{-B\Delta t} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{-A\Delta t} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{-B\Delta t} & \Delta t e^{-B\Delta t} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{-B\Delta t} \end{bmatrix}$$

3.3 SIMULATION SPACE MODEL

Realistic target trajectories are first simulated in three dimensional inertial space. The trajectories are then projected into the FLIR image plane. In order to describe the three dimensional motion in the inertial plane relative to the FLIR plane, a system of coordinate frames and transformations must be defined.

3.3.1 COORDINATE FRAMES

Target Frame - The origin is the target's center of mass. The \underline{e}_v axis lies along the velocity vector. Perpendicular to the first, the second axis, \underline{e}_{pv} points out the right side of the target. The \underline{e}_{ppv} axis completes the right handed coordinate frame pointing out the underside of the fuselage. (v: along velocity vector, pv: perpendicular to velocity vector, ppv: perpendicular to both)

Inertial Frame - The origin is the position of the FLIR sensor. The \underline{e}_x basis vector is the zero azimuth line in the plane tangent to the earth's surface. The orientation of the \underline{e}_x vector in space is arbitrary; it is assumed to point toward the local North for simulation purposes. Perpendicular to \underline{e}_x , the \underline{e}_z vector is the 90 degree azimuth line. The \underline{e}_y basis vector is in the inertial "up" direction, opposite to the gravitational field for a flat Earth approximation. The elevation angle, α , is measured up from the x_I - z_I plane.

α - β -r Frame - The origin is also the center of mass of the target. The \underline{e}_r basis vector is coincident with the true line of sight from the sensor to the target. The α - β plane is defined by the unit vectors \underline{e}_α , \underline{e}_β , which are rotated from the inertial frame \underline{e}_x and \underline{e}_y axis by the amount of the azimuth angle, α , and elevation angle β .

α - β (FLIR) Plane - As was stated previously, the target position is propagated by Equation (3-7) in the α - β plane. This plane is the FLIR image plane in which the sensor makes its measurements. By assuming small azimuth and elevation angles, α and β can be considered the linear translational coordinates x and y . The \underline{e}_y coordinate basis vector is down, with the \underline{e}_x vector to the right, as seen from looking through the FLIR plane to see the target image. This choice of coordinates allows for a right handed system with the distance to target from the FLIR plane measured positive away from the sensor. The x and y coordinates are measured relative to the center of the FLIR plane.

Absolute α - β - r frame - This frame is similar to the α - β - r with the exception that the absolute frame is fixed in inertial space at the initial α - β - r coordinates of the target. This coordinate system uses the true angular arc of the pixels to define target and sensor FOV positions. This frame is used in the simulation to allow for the generation of the target and sensor variables by the truth model.

3.3.2 VELOCITY PROJECTION ONTO THE FLIR PLANE. The true target trajectory is defined in inertial space. The deterministic azimuth velocity, $\dot{\alpha}(t)$, and elevation velocity $\dot{\beta}(t_i)$ are derived from the inertial target velocities. As presented in Loving [4], the inertial velocities are projected into the FLIR image plane based on the geometry in

Figure 3-1 where:

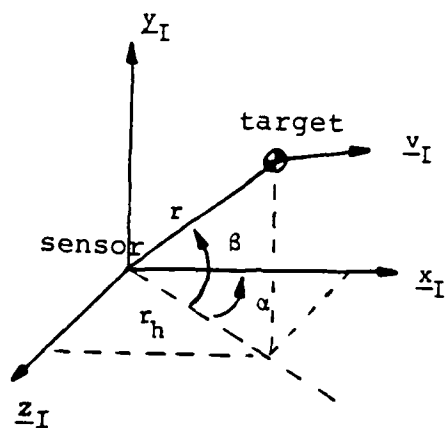


Figure 3-1 Target/Inertial Frame Geometry

x_I, y_I, z_I = inertial axes

r = range from tracker origin to target

r_h = horizontal range

v_I = target inertial velocity

α = azimuth angular displacement

β = elevation angular displacement

The geometry associated with azimuth direction is shown in Figure 3-2:

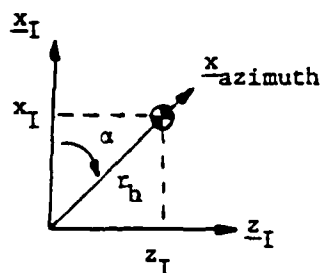


Figure 3-2. Azimuth Geometry

From Figure 3-2,

$$\alpha(t) = \tan^{-1} \frac{z_I(t)}{x_I(t)} \quad (\text{rads}) \quad (3-9)$$

and so

$$\dot{\alpha}(t) = \frac{x_I(t)\dot{z}_I - z_I(t)\dot{x}_I(t)}{z_I^2(t) + x_I^2(t)} \quad (\text{rads/sec}) \quad (3-10)$$

The azimuth velocity from Equation (3-10) is in rads/sec, which must be converted to pixels/sec by dividing by 20×10^{-6} rads/pixel [2,7,8].

Similarly, Figure 3-3 illustrates the geometry involved in computing the elevation velocity.

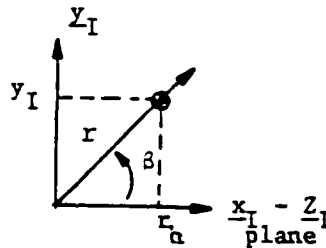


Figure 3-3. Elevation Geometry

where

$$r(t) = \text{range} = [x_I^2(t) + y_I^2(t) + z_I^2(t)]^{1/2}$$

$$r_h(t) = \text{horizontal range} = [x_I^2(t) + z_I^2(t)]^{1/2}$$

and

$$\beta(t) = \tan^{-1} \frac{y_I(t)}{r_h(t)} \quad (\text{rads}) \quad (3-11)$$

Differentiating Equation (3-11) yields:

$$\dot{\beta}(t) = \frac{r_h(t)\dot{y}_I(t) - y_I(t)\dot{r}_h(t)}{r^2(t)} \quad (\text{rads/sec}) \quad (3-12)$$

where, from the previous $r_h(t)$ expression it can be seen that

$$\dot{r}_h(t) = \frac{x_I(t)\dot{x}_I(t) + z_I(t)\dot{z}_I(t)}{r_h(t)} \quad (3-13)$$

Once again, the velocity must be converted to pixels/sec. Equations (3-10) and (3-12) provide the desired deterministic inputs in the FLIR plane. These velocities are included in Equation (3-8) to generate desired true target motion.

3.4 MEASUREMENT MODEL

The FLIR sensor measures the target intensity function projected onto the FLIR image plane. The target intensity function is made up of infra-red radiation from the target and is corrupted by background and sensor noises. At close ranges targets can be well modeled as the sum of bivariate Gaussian functions with elliptical contours [2,7,8]. Figure 3-4 shows the intensity function for a three hot spot target

on the FLIR image plane.

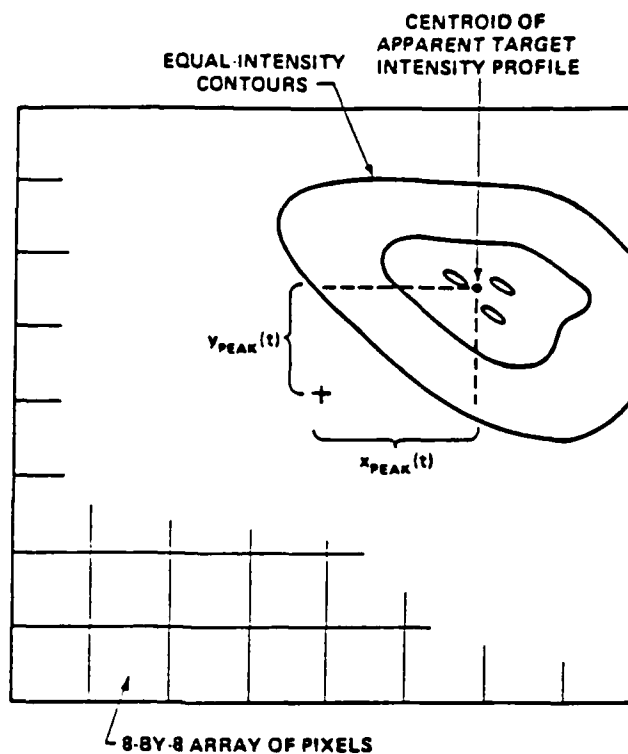


Figure 3-4. Apparent Target Intensity Pattern on FLIR Image Plane

The target used in this effort is composed of three identically distributed hotspots, each of which is described by the following intensity function

$$I(x, y, x_{\text{peak}}(t), y_{\text{peak}}(t)) = I_{\text{max}} \exp\{-.5[(x-x_{\text{peak}})(y-y_{\text{peak}})] \cdot [P_{\alpha\beta}]^{-1}[(x-x_{\text{peak}})(y-y_{\text{peak}})]^T\} \quad (3-14)$$

where

I_{max} = maximum intensity of hotspot

$x_{\text{peak}}, y_{\text{peak}}$ = coordinates of the peak intensity of the hotspot

$E_{\alpha\beta}$ = matrix whose eigenvalues are σ_v^2 and σ_{pv}^2 which are the dispersions of the elliptical constant intensity contours in the target frame, and whose eigenvectors define the orientation of the principle axes.

The x- and y- coordinates in this function are calculated in pixels relative to the center of the FOV.

The location of the hotspots on the target frame are constant. For this simulation the multiple hotspots are arranged as in Figure 3-5.

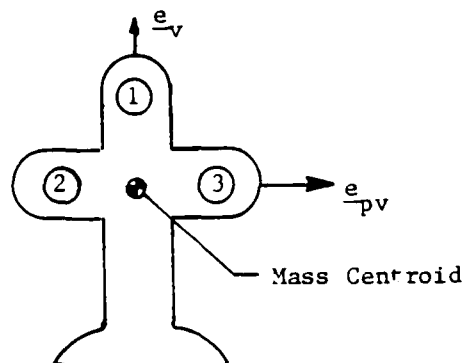


Figure 3-5. Hot Spot Distribution

The hotspot coordinates are:

hotspot	e_v	e_{pv}
1	1. meter	0
2	0.	.5 meters
3	0.	-.5 meters

It is assumed that the velocity vector points out the nose of the target for all time. Additionally it is assumed that the semi-major axis of each intensity function is aligned with the velocity vector. This implies that the target's angle of attack and sideslip angle are negligible

for all time. Although this is not realistic, it simplifies the simulation of the target dynamics, without degradation of the performance analysis.

The intensity measurement seen by each pixel in the 8 x 8 array is the sum of each hotspot's contribution, and the contribution from noise, both background and FLIR. The measurement value for pixel $k1$ (i.e. the k^{th} row and l^{th} column) is:

$$z_{k1}(t_i) = \sum_{m=1}^M \{1/A_p \int_{\text{pixel } k1} I_m[x, y, x_{\text{peak}m}(t_i), y_{\text{peak}m}(t_i)] dx dy\} + v_{k1}(t_i) \quad (3-15)$$

where

M = number of hotspots

$I_m[*]$ = intensity function of the m^{th} hotspot of M total hotspots

$z_{k1}(t_i)$ = output of the $k1^{th}$ pixel (k^{th} row, l^{th} column) at time t_i ; the average intensity at that pixel as sensed by a detector in the FLIR image plane.

A_p = area of one pixel

(x, y) = coordinates of any point within the $k1^{th}$ pixel

$(x_{\text{peak}m}, y_{\text{peak}m})$ = location of the peak of the m^{th} intensity function at t_i

$v_{k1}(t_i)$ = additive FLIR and background noises for the $k1^{th}$ pixel

3.5 TARGET IMAGE PROJECTION ONTO THE FLIR IMAGE PLANE

During the simulation, the true target position is propagated in three dimensional space. This target position must be projected onto the two dimensional FLIR plane in x and y coordinates, along with an associated image on the FLIR plane. Although the image hotspots do not vary with respect to the target frame, they do change orientation with respect to the FLIR plane. In addition, any target motion orthogonal to the FLIR plane must be translated into a size change of the target image as the target's range is changed. To accomplish this, a reference image is defined and variation of this image produced by the following relationships based on the geometry of Figure 3-6.

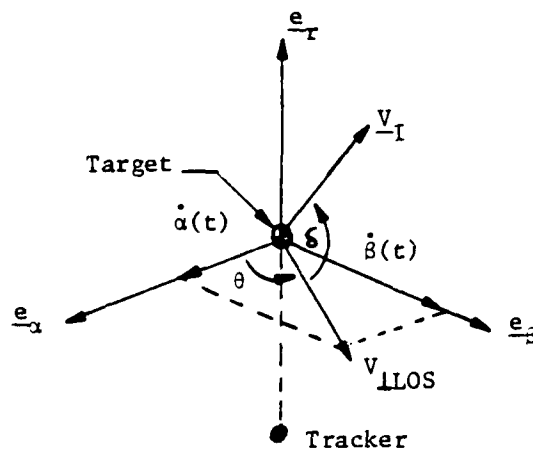


Figure 3-6
Image Projection Geometry

The reference image is oriented to correspond to the

largest planform at a specified range. Changes in sensor/target geometry and range are referenced against this image to create the corresponding current target image.

The current image is then defined by:

$$\sigma_{pv} = \sigma_{pvo} p_0 / p$$

$$\begin{aligned} \sigma_v &= (p_0/p) [\sigma_{pvo} + \cos \delta (\sigma_{vo} - \sigma_{pv})] \\ &= \sigma_{pv} \{1 + [(v \perp \text{LOS})/v_I] [AR - 1]\} \end{aligned}$$

where σ_{vo} , σ_{pvo} = the dispersion of the target along the major and minor axes of the radiating ellipsoid, i.e., axes along and perpendicular to the velocity vector, respectively, for the reference image

σ_v, σ_{pv} = the current dispersions of the target image

p_0 = reference range from sensor to the target

p = current range from sensor to the target

v_I = inertial velocity vector

$v \perp \text{LOS}$ = projection of v_I onto the $\alpha - \beta$ plane, the plane perpendicular to the line of sight to the target

δ = angle between the inertial velocity vector, v_I , and the $\alpha - \beta$ plane, as shown in Figure 3-6

$AR = \sigma_{vo}/\sigma_{pvo}$ = maximum aspect ratio of the hot-spot reference image

The orientation of the hotspots on the FLIR image plane can be obtained from transforming the coordinates from the target frame to the $\alpha - \beta$ plane. From the Figure 3-6 it is seen that

$$\cos \theta = \dot{\alpha}(t) / [v \perp \text{LOS}]$$

$$\sin \theta = \dot{\beta}(t) / [v \perp \text{LOS}]$$

$$\text{where } v \perp \text{LOS} = [\dot{\alpha}(t)^2 + \dot{\beta}(t)^2]^{1/2}$$

The transformation is then accomplished via:

$$\begin{aligned} \underline{x}_{\alpha\beta} = \begin{bmatrix} x \\ y \end{bmatrix}_{\alpha\beta} &= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}_{\text{target frame}} \\ &= \underline{A} \underline{x} \end{aligned} \quad (3-16)$$

The dispersion matrix is transformed using

$$\underline{P}_{\alpha\beta} = \underline{A} \underline{P} \underline{A}^T$$

As seen in Equation (3-14), it is desirable to have $\underline{P}_{\alpha\beta}^{-1}$; the more convenient transformation to be used is then:

$$\underline{P}_{\alpha\beta}^{-1} = \underline{A} [\underline{P}^{-1}] \underline{A}^T \quad (3-17)$$

3.6 TARGET TRAJECTORIES

To evaluate algorithm performance fully, it is necessary to evaluate the tracker against several different realistic target trajectories. This section presents the trajectories currently available in the simulation. These trajectories are designed to provide realistic target behavior with fairly simple models. The basic equations are described in detail by Millner [13].

Trajectory one - This trajectory is a constant-course, straight-and-level fly-by. The inertial velocity lies in the $x_I - y_I$ plane and is constant throughout the simulation. The initial inertial coordinates x_{I0} and z_{I0} can be seen in Figure 3-7. The value of z_I as seen here and Figure 3-8 (for trajectory two) is the parameter varied to determine range sensitivity in Section 6.5.

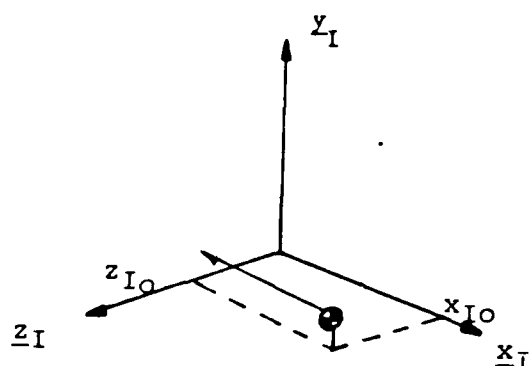


Figure 3-7 Trajectory One

Trajectory two - This trajectory simulates a constant-g pull-up. The target performs a trajectory one maneuver until $t = 2.0$ seconds, allowing the filter to obtain good state estimates before the maneuver is initiated. At time > 2.0 seconds the target performs a constant speed, constant pitch-rate pull-up parallel to the $x_I - y_I$ plane. The maneuver is initiated by instantaneously applying the full maneuver pitch rate as a step change input. Although this creates an artificially harsh maneuver, it allows for easy

implementation in the simulation. Adequate tracking performance against this trajectory should provide good performance against the more benign "real world" trajectories. Figure 3-8 presents the general trajectory two plot. The nominal time for the maneuver to begin is shown as $t=2$ seconds into the simulation. The minimum range cross over point (i.e. where $x_I=0$) is shown. The nominal time for minimum range crossover in this study is $t=5$ seconds.

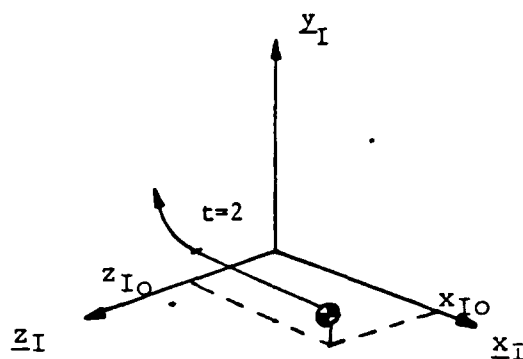


Figure 3-8 Trajectory Two

Trajectory three - This trajectory contains two maneuver changes. Trajectory two is flown until $t = 3.5$ seconds, at which time pitch rate is instantaneously set to zero. Note again this is an artificially harsh simulated maneuver. The inertial velocity vector of the target is maintained through the remainder of the simulation.

Trajectory four - This trajectory parallel to the same as trajectory two, except the maneuver takes place parallel to the x_I - z_I plane, with the "pull-up" about the y_I - axis, rather than the z_I - axis. This maneuver allows for filter evaluation of target motion out of the FLIR plane. In this trajectory the target image changes more dramatically than in the other trajectories.

3.7 SPATIALLY CORRELATED BACKGROUND NOISE

The noise term $v_{k1}(t_i)$ in Equation (3-15) is the spatially correlated background noise. Research conducted by Harnly and Jensen [2,7,8] found this correlation distance to be about two pixels. This was modeled by maintaining non-zero correlation coefficients between each pixel and its two closet neighbors. In the 8×8 tracking window (FLIR plane) the 64 pixel elements make up a 64-element vector, and thus the spatially correlated noise can be modeled as:

$$\underline{v}(t_i) = \sqrt{C} \sqrt{R} \underline{v}'(t_i)$$

where $\underline{v}'(t_i)$ = a 64 - dimensional vector of independent discrete-time white Gaussian noise processes with statistics:

$$E(\underline{v}'(t_i)) = \underline{0}$$

$$E(\underline{v}'(t_i)\underline{v}'(t_j)) = \underline{I}\delta_{ij}$$

The process $\underline{v}(t_i)$ has covariance:

$$E(\underline{v}(t_i)(\underline{v}^T(t_j))) = R \delta_{ij}$$

where R is the 64×64 matrix of correlation coefficients. The detailed development of the spatially correlated noise using non-zero spatial correlation for the nearest neighbor pixels is presented in Harnly and Jensen [2,7,8] and Kozemchak [3].

3.8 SUMMARY

This chapter presents the truth model representation of the real world used in the computer simulation to evaluate the tracker's performance. Truth model state propagation in inertial space and the transformations required to project target motion onto the FLIR image plane are provided. In addition, the measurement model is developed, including the measurement noise statistics.

IV. TRACKING ALGORITHM

4.1 Introduction

The linear Kalman filter/correlator is the basic element in the MMAF discussed in Chapter 2. This chapter describes the filter/correlator as it is applied to the MMAF in this effort. Filter propagation, measurement update, template creation and correlation are presented. Additionally, the FLIR sensor pointing controller development used to model control system lags is developed in this chapter.

4.2 State Space Model

The linear Kalman filter uses an eight-dimensional state space model. The filter states are target position (x_D , y_D), velocity (v_x , v_y), acceleration (a_x , a_y), and atmospheric jitter position (x_A , y_A), in the FLIR image plane. The atmospheric model used here is first order, as opposed to the third order truth model discussed in Chapter 3. The model is reduced in order by neglecting the high frequency double pole. In addition, the target acceleration and atmospheric position jitter are modeled as first order Gauss-Markov processes.

The filter structure is identical to that presented in Loving [4].

$$\mathbf{x}_F = [x_D \ y_D \ v_x \ v_y \ a_x \ a_y \ x_A \ y_A]^T \quad (4-1)$$

The state equations are

$$\begin{aligned} \dot{x}_D &= v_x \\ \dot{y}_D &= v_y \\ \dot{v}_x &= a_x \\ \dot{v}_y &= a_y \\ \dot{a}_x &= (-1/\tau_{DF})a_x + w_{Dx} \\ \dot{a}_y &= (-1/\tau_{DF})a_y + w_{Dy} \\ \dot{x}_A &= (-1/\tau_{AF})x_A + w_{Ax} \\ \dot{y}_A &= (-1/\tau_{AF})y_A + w_{Ay} \end{aligned} \quad (4-2)$$

where

τ_{DF} = correlation time for target acceleration

τ_{AF} = correlation time for atmospheric jitter

$w_{Dx}, w_{Dy}, w_{Ax}, w_{Ay}$ = zero-mean white Gaussian noise processes
whose strengths depend on tuning
results

Identical, independent models are used to represent the states in the x- and y-directions of the FLIR image plane.

The above relationships can be written as a state vector

differential equation in standard form. It is noted that no deterministic control term, $E_d u(t_1)$, is included in Equation (4-2). The control application will be discussed in Section 4.4.

$$\dot{\underline{x}}_F(t) = \underline{F}_F \underline{x}_F(t) + \underline{G}_F \underline{w}_F(t) \quad (4-2)$$

\underline{F}_F = The time invariant system plant matrix which is:

$$\underline{F}_F = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1/\tau_{DF} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/\tau_{DF} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_{AF} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_{AF} \end{bmatrix}$$

\underline{G}_F = the time invariant system noise output matrix

$$\underline{G}_F = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\underline{w}_F(t) = [w_{Dx}, w_{Dy}, w_{Ax}, w_{Ay}]^T$, the noise vector of mutually independent zero-mean white Gaussian noise processes with:

$$E(\underline{w}_F(t) \underline{w}_F^T(t + \tau)) = \underline{Q}_F \delta(\tau)$$

$$Q_F = \begin{bmatrix} 2\sigma_{DF}^2/\tau_{DF} & 0 & 0 & 0 \\ 0 & 2\sigma_{DF}^2/\tau_{DF} & 0 & 0 \\ 0 & 0 & 2\sigma_{AF}^2/\tau_{AF} & 0 \\ 0 & 0 & 0 & 2\sigma_{AF}^2/\tau_{AF} \end{bmatrix}$$

σ_{DF}^2 = target acceleration variance and mean squared value

σ_{AF}^2 = atmospheric jitter position offset variance and mean squared value

The equivalent discrete time equation of Equation (4-3) is [5]:

$$\underline{x}_F(t_{i+1}) = \underline{\Phi}_F(t_{i+1}, t_i) \underline{x}_F(t_i) + \sqrt{Q_F} \underline{w}_{Fd}(t_i) \quad (4-3)$$

where

$\underline{x}_F(t_i)$ = filter state vector at time t_i

$\underline{w}_{Fd}(t_i)$ = discrete-time zero-mean white Gaussian noise of covariance = \underline{I}

The state transition matrix is [16]:

$$\underline{\Phi}_F(t_{i+1}, t_i) = \begin{bmatrix} 1 & 0 & \Delta t & 0 & J1 & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & J1 & 0 & 0 \\ 0 & 0 & 1 & 0 & J2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & J2 & 0 & 0 \\ 0 & 0 & 0 & 0 & J3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & J3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & J4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & J4 \end{bmatrix}$$

where

$$J1 = \tau_{DF}[\Delta t - \tau_{DF}(1 - \exp(-\Delta t/\tau_{DF}))]$$

$$J2 = \tau_{DF}[1 - \exp(-\Delta t/\tau_{DF})]$$

$$J3 = \exp(-\Delta t/\tau_{DF})$$

$$J4 = \exp(-\Delta t/\tau_{AF})$$

$$\Delta t = t_{i+1} - t_i$$

The growth in uncertainty due to dynamic driving noise is described by [13]:

$$Q_{Fd} = \begin{bmatrix} Q_{11} & 0 & Q_{13} & 0 & Q_{15} & 0 & 0 & 0 \\ 0 & Q_{11} & 0 & Q_{13} & 0 & Q_{15} & 0 & 0 \\ Q_{13} & 0 & Q_{33} & 0 & Q_{35} & 0 & 0 & 0 \\ 0 & Q_{13} & 0 & Q_{33} & 0 & Q_{35} & 0 & 0 \\ Q_{15} & 0 & Q_{35} & 0 & Q_{55} & 0 & 0 & 0 \\ 0 & Q_{15} & 0 & Q_{35} & 0 & Q_{55} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Q_{77} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{77} \end{bmatrix}$$

where

$$Q_{11} = \sigma_{DF}^2 \left[2\tau_{DF}\Delta t^3/3 - [2\tau_{DF}^2\Delta t^2] - [4\tau_{DF}^3\Delta t \exp(-\Delta t/\tau_{DF})] \right. \\ \left. + [2\tau_{DF}^3\Delta t] - [\tau_{DF}^4 \exp(-2\Delta t/\tau_{DF})] + \tau_{DF}^4 \right]$$

$$Q_{13} = \sigma_{DF}^2 \left[\tau_{DF}\Delta t^2 + [2\tau_{DF}^2\Delta t \exp(-\Delta t/\tau_{DF})] + [\tau_{DF}^3] \right. \\ \left. - [2\tau_{DF}^3 \exp(-\Delta t/\tau_{DF})] - [2\tau_{DF}^2\Delta t] \right. \\ \left. + [\tau_{DF}^3 \exp(-2\Delta t/\tau_{DF})] \right]$$

$$Q_{15} = \sigma_{DF}^2 \left[-2\tau_{DF}\Delta t \exp(-\Delta t/\tau_{DF}) + [\tau_{DF}^2] \right. \\ \left. - [\tau_{DF}^2 \exp(-2\Delta t/\tau_{DF})] \right]$$

$$Q_{33} = \sigma_{DF}^2 \left[2\tau_{DF}\Delta t - [3\tau_{DF}^2] + [4\tau_{DF}^2 \exp(-\Delta t/\tau_{DF})] \right. \\ \left. - [\tau_{DF}^2 \exp(-2\Delta t/\tau_{DF})] \right]$$

$$Q_{35} = \sigma_{DF}^2 \left(\tau_{DF} - [2\tau_{DF} \exp(-\Delta t/\tau_{DF})] + [\tau_{DF} \exp(-2\Delta t/\tau_{DF})] \right)$$

$$Q_{55} = \sigma_{DF}^2 (1 - \exp(-2\Delta t/\tau_{DF}))$$

$$Q_{77} = \sigma_{AF}^2 (1 - \exp(-2\Delta t/\tau_{AF}))$$

Using the terms defined above, the state estimate and covariance are propagated as follows:

$$\hat{\mathbf{x}}(t_{i+1}^-) = \mathbf{F}(t_{i+1}, t_i) \hat{\mathbf{x}}(t_i^+) \quad (4-5)$$

$$\mathbf{P}(t_{i+1}^-) = \mathbf{F}(t_{i+1}, t_i) \mathbf{P}(t_i^+) \mathbf{F}^T(t_{i+1}, t_i) + \mathbf{Q}_F d \quad (4-6)$$

where

$\hat{\mathbf{x}}(t_i^+)$ = the state estimate after update at time t_i

$\hat{\mathbf{x}}(t_{i+1}^-)$ = the state estimate prior to update at time t_{i+1}

$\mathbf{P}(t_i^+)$ = the conditional covariance matrix after update at time t_i

$\mathbf{P}(t_{i+1}^-)$ = the conditional covariance matrix prior to update at time t_{i+1}

As is noted from Equation (4-5) no control term is included in the state propagation equation. This is due the manner in which the control is applied. The FLIR sensor is pointed at the filter's propagated position estimate prior to the measurement being taken (assuming an ideal controller). The measurements and filter states are then relative to the center of the FOV. Control application is discussed in detail in Section 4.4.

4.3 MEASUREMENT MODEL

The linear Kalman filter algorithm uses pseudo-measurements to allow the linear filter structure. As discussed in [4,10,14,16], the need for implementation of the extended Kalman filter was due to the non-linear measurements obtained from the FLIR sensor. By providing x and y offsets from an enhanced correlator to the filter as pseudo-measurements, a linear Kalman filter structure is

appropriate. The development in this section assumes an ideal controller as discussed in Section 1.2.

4.3.1 FILTER PROCESSING IN THE FOURIER DOMAIN

Many of the operations required to perform the target image estimation are more easily accomplished in the Fourier domain. Additionally, Fourier transforms lend themselves well to optical implementation, reducing the computer resources required. For these reasons the discrete two-dimensional Fourier transform (DFT) is used on information in the FLIR image plane. The filter processing in the Fourier domain is depicted in Figure 1-1. A rigorous mathematical description of the Fourier transforms as applied to the filter algorithm is presented in [4,10,14,16].

4.3.1.1 TARGET IMAGE ESTIMATION (TEMPLATE)

As was discussed in Section 1.2.1, the linear Kalman filter pseudo-measurements are provided by correlating the expected target image with that obtained during the current sensor measurement. To accomplish this, the algorithm estimates the target image by creating a template composed of an average of previous target intensity profiles. To

incorporate the latest intensity profile into this average, the current image must first be centered on the FLIR image plane. This is required to overlay the current intensity function on the template intensity function which is maintained at the center of the FLIR FOV. This is accomplished using the Fourier transform shifting property. The target intensity function is shifted on the FLIR image plane, relative to the center of the FOV, by the amount;

$$\hat{x}_{\text{shift}}(t_i^+) = \hat{x}_D(t_i^+) + \hat{x}_A(t_i^+) \quad (4-7)$$

and similarly in the y FLIR direction. The values of $\hat{x}_D(t_i^+)$ and $\hat{x}_A(t_i^+)$ are obtained from the Kalman filter state estimates. With the current image intensity function overlayed on the template function, the current image is incorporated into the template by exponential smoothing. As previously stated, it is assumed here that measurement corruption and background noises will vary more rapidly than target intensity profiles from sample to sample. This assumption is exploited by the use of an exponential smoothing algorithm. Exponential smoothing approximates a true finite memory average while requiring only storage of the current template. The equation is

$$\hat{y}(t_i) = \alpha y(t) + (1-\alpha) \hat{y}(t_{i-1}) \quad (4-8)$$

$\hat{y}(t_i)$ = current "average" or smoothed value

$y(t_i)$ = current data frame

$\hat{y}(t_{i-1})$ = previous average or smoothed value

α = smoothing constant, $0 < \alpha \leq 1$

The appropriate value of α is dependent on the target dynamics. A rapidly changing target requires a heavier weighting of more recent frames, thus a higher value of α . Based on previous studies by Sulzu [16] and Loving [4], the value of $\alpha = .1$ is used for this effort.

4.3.1.2 PSEUDO-MEASUREMENTS BY ENHANCED CORRELATION

The pseudo-measurements used by the linear Kalman filter are linear x and y FLIR-plane offsets of the current target image, relative to the center of the FLIR FOV. These offsets are determined by the enhanced correlator. The correlator is considered enhanced since it compares current intensity functions with a template, as opposed to the previous measurement intensity function. The template centroid is maintained at the center of the FLIR's FOV. The cross correlation is performed by taking the inverse Fourier transform (IFFT) of the relationship of Equation (4-9)

$F[g(x,y)] = G(f_x, f_y) \Rightarrow$ transform of FLIR data

$F[l(x,y)] = L(f_x, f_y) \Rightarrow$ transform of template

$$[g(x,y) * l(x,y)] = G(f_x, f_y) \times L^*(f_x, f_y) \quad (4-9)$$

where

$[g(x,y) * l(x,y)]$ = cross correlation
of the $g(x,y)$ and $l(x,y)$

$L^*(f_x, f_y)$ = complex conjugate of the Fourier transform of
the sequence $l(x,y)$

The correlation between the template and the current target image produces a measurement which can be written as

$$x_{\text{offset}}(t_i) = [x_{Dt}(t_i) + x_{At}(t_i)] - \hat{x}_{Dmmaf}(t_i^-) + v(t_i) \quad (4-10)$$

where

$x_{Dt}(t_i)$ = the true x position of the target centroid

$x_{At}(t_i)$ = the true atmospheric distortion in the x coordinate

$v(t_i)$ = the measurement noise described in Section 4.3.2

$\hat{x}_{Dmmaf}(t_i^-)$ = is the multiple model adaptive filter
x coordinate position due to dynamics

where the filter states are in absolute FLIR coordinates.

Equation (4-10) assumes an ideal controller/pointer: the center of the FOV is positioned at $\hat{x}_{Dmmaf}(t_i^-)$ after control is applied. Equation (4-10) can be written as

$$x_{\text{offset}}(t_i) = x_{Dt}(t_i) + x_{At}(t_i) + v(t_i) \quad (4-11)$$

the absolute coordinate frame discussed in Chapter 3 to the current center of the FOV. The target image position is then measured relative to the center of the FOV, established by the current pointing of the sensor, through the action of the controller.

Equation (4-11) is in the form of a standard Kalman filter measurement equation;

$$\underline{z}(t_i) = \underline{H}(t_i) \underline{x}(t_i) + \underline{v}(t_i) \quad (4-12)$$

which is discussed in the next section.

4.3.2 KALMAN FILTER UPDATE EQUATIONS

The linear Kalman filter update equations are of the standard form as presented in [4,5]. The appropriate measurement equation is

$$\underline{z}(t_i) = \underline{H}_F(t_i) \underline{x}(t_i) + \underline{v}_F(t_i) \quad (4-13)$$

where

$\underline{z}(t_i)$ = the offset estimate in x- and y-coordinates produced by the correlator, based on the filter-predicted centroid location (i.e., the vector sum of the dynamics and atmospheric states)

$$= \begin{bmatrix} x_{AD}(t_i) + x_{AC}(t_i) \\ y_{AD}(t_i) + y_{AC}(t_i) \end{bmatrix} + \begin{bmatrix} v_{F1}(t_i) \\ v_{F2}(t_i) \end{bmatrix}$$

$$\underline{H}_F = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$\underline{v}_F(t_i)$ = noise produced by the correlation algorithm with statistics that were shown empirically to be [10,13,14]:

$$E\{\underline{v}_F(t_i)\} = \underline{0}$$

$$E\{\underline{v}_F(t_i) \underline{v}_F^T(t_j)\} = \underline{R}_F(t_i) \delta_{ij}$$

where

$$R_F = \begin{bmatrix} .00436 & 0 \\ 0 & .00598 \end{bmatrix}$$

The standard Kalman filter update equations apply:

$$K(t_i) = P_F(t_i^-) H_F^T [H_F P_F(t_i^-) H_F^T + R_F]^{-1} \quad (4-14)$$

$$\hat{x}_F(t_i^+) = \hat{x}_F(t_i^-) + K(t_i)(z(t_i) - H_F \hat{x}_F(t_i^-)) \quad (4-15)$$

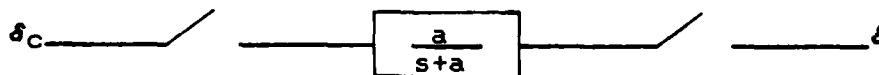
$$P_F(t_i^+) = P_F(t_i^-) - K(t_i) H_F P_F(t_i^-) \quad (4-16)$$

where all the terms have been previously defined.

4.4 FLIR SENSOR POINTING CONTROLLER

The pointing controller is tasked to position the FLIR sensor on command to the filter's propagated estimate of the target's position. This task is ideally to be accomplished in less than one sample period. However, servo lags inherent in mechanisms with significant inertia will cause less than ideal pointing performance. If the filter is not aware of this imperfect pointing at the time of a measurement update, the filter may interpret the offset due to non-ideal pointing as target motion and provide inaccurate estimates of future states.

Assume a deterministic first order controller lag of the following form:



where the input, a commanded change in pointing position

denoted as δ_c , is the propagated estimate of the target's position $\hat{x}_D(t_i^-)$. The output, δ , is the actual change in pointing position of the FLIR sensor. The controller lag is assumed to model the time response of the controller algorithm dynamics as well as time lags inherent in the pointing servo. It is seen by the form of the lag transfer function that, given a sufficiently long sample period for a fixed controller time constant, the controller servo would be able to point the FLIR sensor at the desired position. However, long sample periods provide less frames of data per second, which reduces filter performance.

4.4.1 POINTING_CONTROLLER_IMPLEMENTATION

The pointing command is the change in position on the FLIR image plane, from the current center of the FOV to the propagated estimate of the target's position on the FOV. This command is modeled as a step input in both the x and y directions of the FLIR plane. In the Laplace domain, the output for a first order controller with a step input is:

$$\frac{\delta(s)}{\delta_c(s)} = \frac{a}{s + a}$$

where

δ_c = controller commanded step change

a = controller time constant

δ = control actually applied

The change in output position attained one sample period after the commanded change of $\delta(t_{i+1})$ is received at time t_i , is then

$$\delta(t_{i+1}) = \delta_C(t_{i+1}) [1 - e^{-a\Delta t}] \quad (4-17)$$

where t is the sample period. The output of Equation (4-17) is the actual movement of the FLIR sensor in the FLIR plane. However the filter is unaware of the true FLIR position, assuming instead that the center of the FOV is positioned at $\hat{x}_{Dmmaf}(t_{i+1}^-)$.

The measurement relative to the FLIR center of view is then:

$$z(t_i)_k = [x_{Dt}(t_i^-) + x_{At}(t_i^-) + r(t_i)]_k - \hat{x}_{Dmmaf}(t_{i+1}^-) + v(t_i) \quad (4-18)$$

where the states are in absolute FLIR coordinates.

$z(t_i)_k$ = measurement of the k^{th} filter

$x_{Dt}(t_i^-)$ = position due to true target dynamics

$x_{At}(t_i^-)$ = value of the true atmospheric jitter state

$\hat{x}_{Dmmaf}(t_{i+1}^-)$ = propagated estimate of the dynamic states from the MMAF (pointing command)

$r(t_i)$ = the error due to controller lags of the k^{th} filter

The additional offset included in the measurement, $r(t_i)$, is interpreted by the filter as an error in the propagated estimate of the states. The filter then adjusts the state

estimate of the states. The filter then adjusts the state estimates to account for the "additional error", creating an actual error in the state estimate.

It should be noted here, that the model used in this effort simulates a filter with measurements based on coordinates relative to the FLIR FOV. If a filter based on absolute FLIR coordinates, as described in Section 3.3.1, were implemented, an estimate of absolute sensor position would be required to perform the shift operation used in template generation.

The controller dynamics could be included in the filter to provide the filter knowledge of the pointing errors. This however would be at the expense of additional computation time. The objective of this study is to analyze the effects of pointing lags on a filter that assumes there are none.

4.5 FILTER IMPLEMENTATION

As was discussed in Chapter 2, the MMAF is made up of a bank of 3 elemental filters. Each filter is tuned for an independent tracking scenario by varying the dynamics model correlation time, τ_{DF} , and parameter σ_{DF} that determines the dynamic driving noise Q_{DF} (described in Section 4.2), as well as the FOV.

The specific parameter values, for each filter, as established by Loving [4] are presented in Table 4-1:

Table 4-1

Filter	Trajectory	FOV	τ_{DF}	σ_{DF}^2
1	Benign	Small	3.5	1000
2	20 g	Large	1.5	2000
3	10 g	Small	1.5	1500

The FOV size corresponds to the size of the pixels used in the filter's FOV.

As was discussed in Section 2.3, a center 8 x 8 pixel grid is padded by 8 rows and columns of data on all sides, creating a 24 by 24 pixel matrix for performing the FFT's. For the small FOV, the center 8 x 8 grid is made up of pixels 20 micro-rads x 20 micro-rads. This inner frame is padded with noise-corrupted data to reduce the high frequency effect of edges where the target image "spills over" the center 8 x 8 array [10,14].

The large FOV is created by taking the 24 x 24 pixel matrix and using each 3 x 3 block of 20 micro-rads pixel to form one "large" 60 micro-rad x 60 micro-rad pixel. The intensity value of the "small" 20 micro-rad pixels are averaged to establish the intensity of the "large" pixel. Since the correlation output is in terms of pixels, the large FOV of variables must be adjusted by a factor of 3 in the filter algorithm. This is padded with zeros instead of data when performing the FFT's, since it is assumed that the

target image is usually contained well within the 24 x 24 array of pixels.

The filter contains a reacquisition cycle to allow a filter which has lost track to reacquire the target. The function of the reacquisition cycle is to allow an elemental filter whose errors are diverging to be brought "back on line" to be used in the event it would provide an adequate "match" to the observed target dynamics. For the Bayesian MMAF, the divergent filter states reset to a combination of the non-divergent filter's states. Likewise the covariance matrix is reset to a combination of the non-divergent filter's covariance. The conditional probabilities are left at current values. For the MAP MMAF, the state and covariance of any divergent filter are reset to the current MMAF values.

4.6 SUMMARY

This chapter has presented the linear Kalman filter/enhanced correlator algorithm as was used in this effort. The Kalman filter equations consisting of the state space model for propagation as well as the measurement update equations have been discussed. The FLIR sensor pointer/controller development and implementation has also been included in this chapter.

V. ALGORITHM TEST SET-UP

5.1 INTRODUCTION

This chapter presents the evaluation tools for the investigations made during this effort. The first section covers the derivation of the tracker error statistics. The tracker state estimate errors are of primary interest in evaluating the algorithm's performance. The next section discusses the format of the performance plots which are the main evaluation tool. The greyscales, discussed in Section 5.4, are used to gain insights into the tracker characteristics. The nominal filter parameters are presented in Section 5.5. The final section presents the designation code used in the identification of the performance plots in the appendices.

5.2 TRACKER STATISTICS

The tracker performance statistics were obtained by means of a Monte Carlo analysis technique. Research conducted by Harnly and Jensen [2,7,8], Flynn [1], and Mercier [9,12], have shown that the sample statistics corresponding to ten Monte Carlo runs will exhibit sufficient convergence to the actual error statistics of an infinite number of runs, that ten run Monte Carlo

simulations are used through this research. Tracking simulations of five and ten seconds real time were conducted, generating 150 and 300 frames of data at 30 Hz. sample rate.

The sample mean errors of the filter estimates are calculated as follows [16]:

$$\overline{E_{xd}}(t_i) = 1/N \sum_{k=1}^N [x_{dk}(t_i) - \hat{x}_{dfk}(t_i)] = 1/N \sum_{k=1}^N e_{xdk}(t_i) \quad (5-1)$$

where

$\overline{E_{xd}}(t_i)$ = sample mean error (i.e. ensemble average error over all n simulations) in x-dynamics position at time t_i

$\hat{x}_{dfk}(t_i)$ = multiple model filter estimated x-dynamics value at time t_i for simulation k

$x_{dk}(t_i)$ = truth model x-dynamics value at time t_i for simulation k

$e_{xdk}(t_i)$ = error in x-dynamics position at time t_i for simulation k

N = number of Monte Carlo runs

The sample variance of the error is given by:

$$\sigma_{xd}^2(t_i) = [1/(N-1)] \sum_{k=1}^N e_{xdk}^2(t_i) - [N/(N-1)] \overline{E_{xd}}^2(t_i) \quad (5-2)$$

Two filter error parameters are of special interest in the evaluation. The first is the error committed in estimating

the target position. The second is the error committed in estimating the centroid of the target images on the FLIR plane. Both parameter's statistics are determined for the propagation (t_i^- or "minus"), and measurement update (t_i^+ or "plus") estimates, in the FLIR x and y directions. The errors are measured in pixels relative to the center of the FOV, a pixel being 20 micro-radians on a side.

Temporal averaging was also conducted on the mean error and standard deviation time histories. Time averaging provides a compact tabular evaluation of filter performance. For trajectories one and two, the initial averaging was conducted between $t = 0.5$ and $t = 2.0$ seconds. This time interval was chosen as to allow initial filter transients to die out, while not encountering the nominal minimum range/maximum passing rate condition. This condition occurs due to the target passing the sensor in the inertial x direction. A second temporal averaging interval was chosen from $t = 3.5$ to $t = 5.0$ seconds for maneuvering trajectories. This interval allows most of the transients due the to maneuver at $t = 2.0$ sec. to die out before averaging begins. Care must be exercised in interpreting the time average statistics, as misleading figures of merit can result. For example, if the errors should follow a ramp which reverses sign during the time interval, an insignificant mean error will result. This could lead to the erroneous assumption that an unbiased estimate was being

generated by the filter. For this reason, the performance plots and image greyscales are the major evaluation tools in this study.

5.3 PERFORMANCE PLOTS FORMAT

The graphical representation of the filter performance plots provides a descriptive format from which to evaluate the algorithm. The plots are of x- and y- dynamics state mean error and x- and y- centroid mean error, plus or minus one standard deviation. Both propagated (minus) and updated (plus) error estimates are plotted. Figures 5-1 and 5-2 present the filter y- axis dynamic state estimates prior to (minus) and after (plus) measurement update, plotted against real time. At t=2 seconds, a 10-g pull-up maneuver begins. The increase in filter error, seen shortly afterward, is due to filter's time response in adapting to the perceived change in target trajectory. It is noted that the error standard deviation is smaller for the "plus" estimate than for the "minus", demonstrating the expected reduction in error after measurement incorporation. Figure 5-3 presents the y-axis true rms error plotted with the filter's own estimate of its y- rms error. The upper trace in this figure is the filter computed rms error, with the lower trace being the actual rms error. This plot is a good indication of how well the filter is tuned. The increased

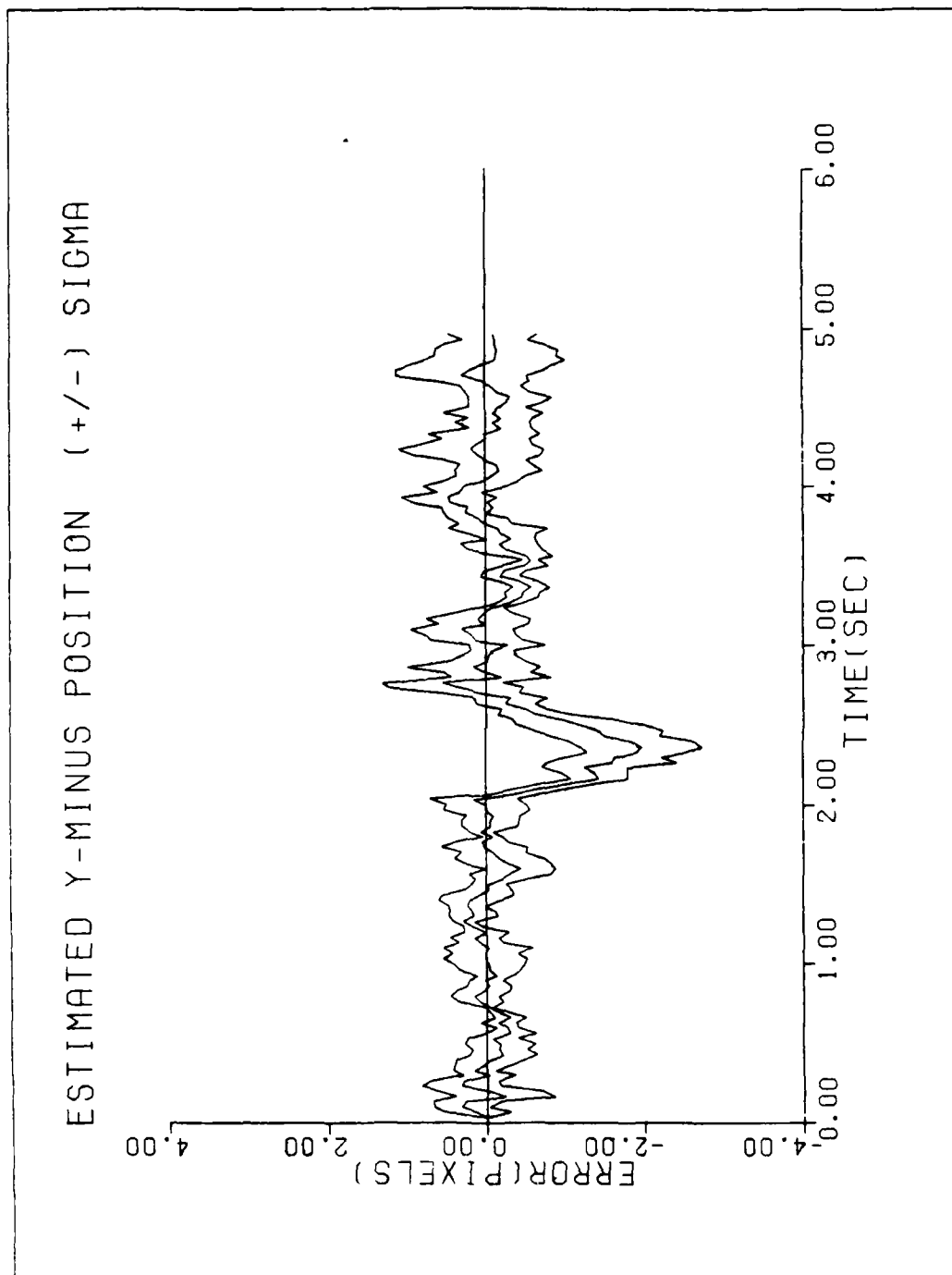


Figure 5-1. Typical Plot - Y Dynamics Before Update

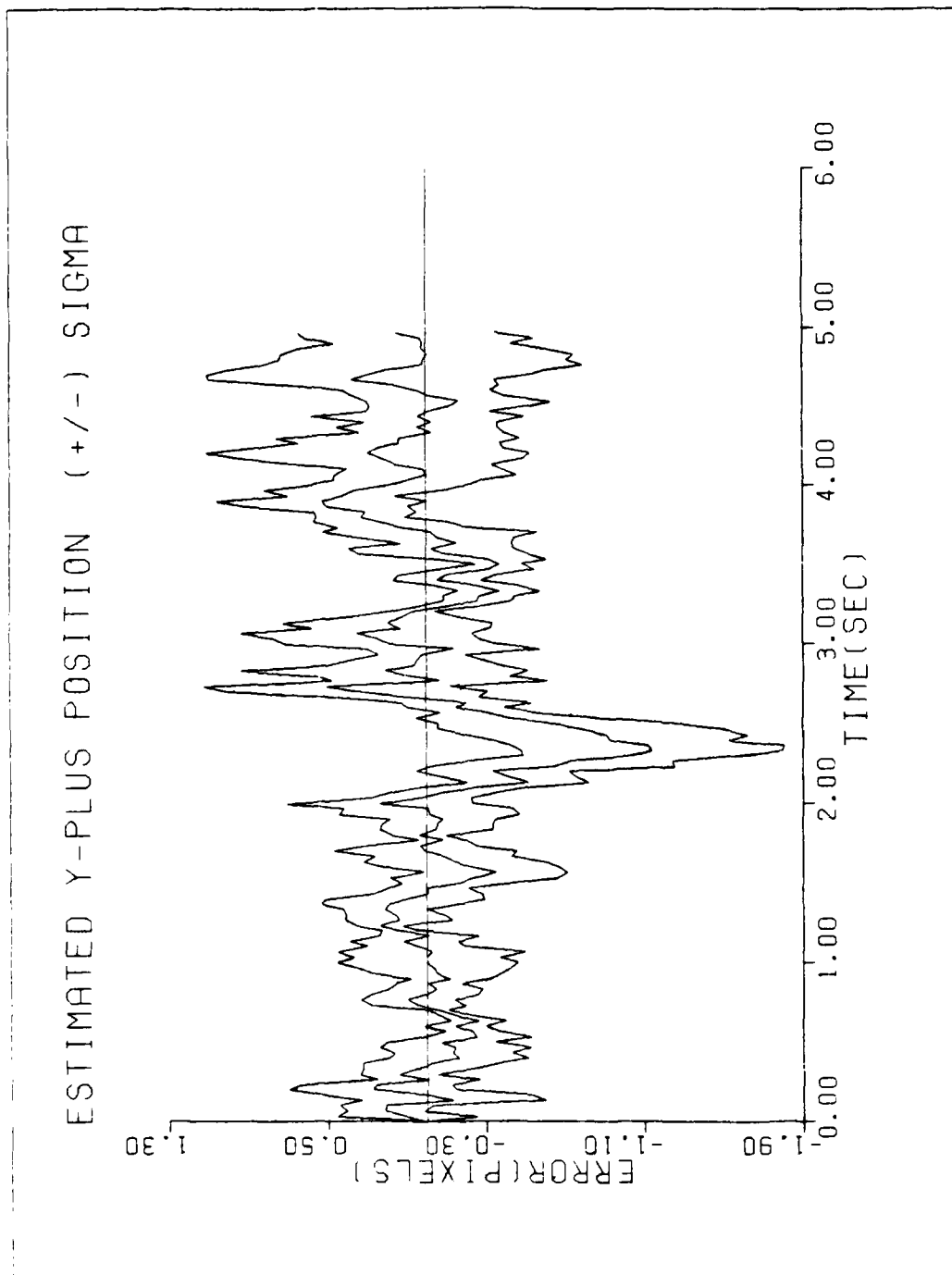


Figure 5-2. Typical Plot - Y Dynamics After Update

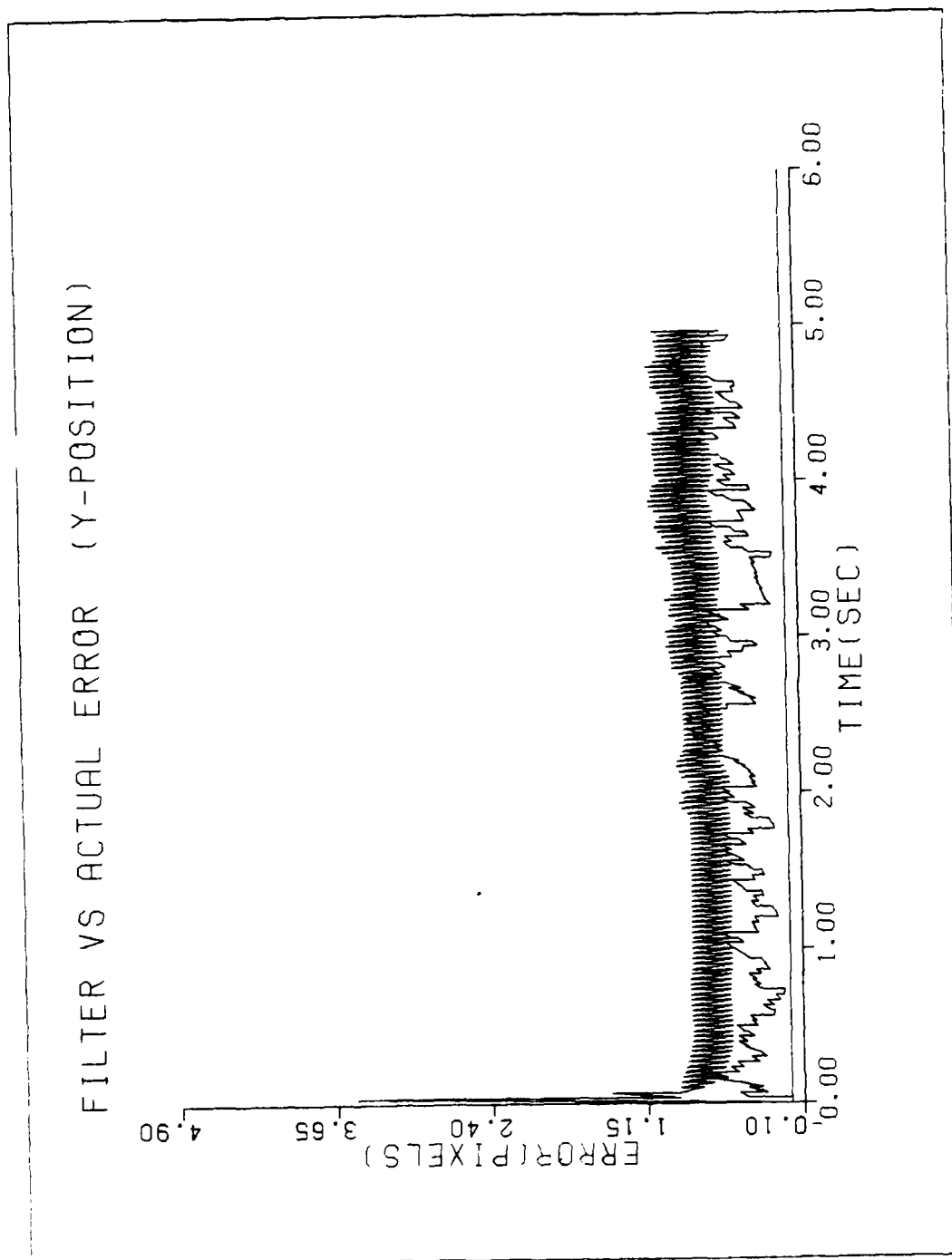


Figure 5-3. Typical Plot - Filter rms Errors vs True rms Error

actual error at approximately $t = 2$ seconds demonstrates the increased filter error as a result of the maneuver. The corresponding increase in filter uncertainty is noted in the filter's computed rms error.

5.4 GREYSCALE

A greyscale is a pictorial display of an image in which shading of the image is used to indicate similar parameters. In this effort, the greyscale figures present the 24 x 24 pixel FOV, with either the target image or template indicated by the shaded region. Here the "darker" shaded regions indicate a higher intensity, the darkness of a pixel being defined by the fullness of a character in the greyscale. For this effort the following scale applies:

Symbol	Indicates	Intensity units
-		$0 \leq I < 5$
+		$5 \leq I < 10$
o		$10 \leq I < 15$
%		$15 \leq I < 20$
X		$20 \leq I < 25$
*		$25 \leq I < 30$
#		$30 \leq I < 35$
@		$35 \leq I < 40$
#		$40 \leq I < 45$
		$I > 50$

A typical greyscale plot is presented in Figure 5-4. The top array shows an offcentered target image. The increased intensity at the image center is visible. Above the array,

RELATIVE PROBABILITIES = .9800, .0100, .0100
 FRAME NUMBER = 2
 123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
123456789012345678901234

-----+-
 -+0*%0-
 +%X##%-
 -0*%XX*+
 ++000+
 --

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

 -++0*0+-
 +0*%XX*-
 +0*%XX*-
 -+0*%*+-
 --++-

Figure 5-4. Greyscale Target Image and Template

the frame number and relative probabilities, p_k , are presented, p_1 , p_2 , and p_3 respectively. For this effort all greyscales pertain to small pixels, 20-micro rads on a side. It is noted that although the array is 24 x 24, the small FOV comprise only the center 8 x 8 grid; the remaining pixels are the FOV padding discussed in Chapter 3. The entire array in the greyscale figures are of interest for a large FOV filter.

5.5 PARAMETER VALUES

In this effort many of the parameter values that appear in the equations discussed in earlier chapters are varied to identify their effect on filter performance. This section defines the nominal values of these parameters used as a baseline in this research. Most of these values are the results of previous efforts, and in many cases were established to provide the "best" performance.

5.5.1 TRUTH MODEL PARAMETERS. The nominal initial conditions on all target trajectories in inertial space are:

Inertial position: $x_I = 5000m$

$y_I = 500m$

$z_I = 20000m$

Inertial velocity: $v_{xI} = -1000\text{m/sec}$

$v_{yI} = 0.$

$v_{zI} = 0.$

Inertial acceleration: $a_{xI} = 0.$

$a_{yI} = 0.$

$a_{zI} = 0.$

The maximum intensity, I_{\max} , of each of the three hotspots is 20 intensity units. The rms value of the (spatially correlated) background and FLIR noises, v_{k1} , used in Equation (3-14), is one (intensity unit²). The corresponding signal to noise ratio, S/N, is then 20. This value is representative of many tracking environments [2,7,8].

For a glint dispersion parameter of $2.0(\text{pixels})^2$, and aspect ratio of one, the resultant eigenvalues of the dispersion matrix, \underline{P} , in the target frame are both 2.0. The atmospheric jitter rms value, i.e., the variance of x_A and y_A is $.2(\text{pixels})^2$. These parameters define the truth model discussed in Chapter 3.

5.5.2 FILTER PARAMETER VALUES

As was discussed in Chapter 4, each elemental filter was tuned to provide the best individual performance against its specific trajectory. These tuning values are presented in Table 4-1. In addition, the atmospheric noise variance used here was $.2(\text{pixels})^2$, with an atmospheric jitter correlation time, τ_{AF} , of 0.07 seconds.

The initial hypothesis conditional probabilities are 0.98, 0.01, 0.01 for filters one through three, respectively. The lower bound on these probabilities is .001.

The ideal controller was used in the filter for all runs except where stated. This controller assumes perfect pointing of the FLIR/laser in less than one sample period.

5.6 DESIGNATION CODE

To facilitate identification of the simulation parameters in the performance plots, a designation code was established. This code consists of 3 major sections, as presented below:

T2	G10	C.995
1	2	3

1. Trajectory Type (T1,T2,T3,T4; see Section 3.6)
2. Maneuver g level (10, 20); this identifier is eliminated for trajectory 1
3. Parameter identifier; this identifier assumes nominal parameter values where an identifier is not specified. The # symbol following an identifier indicates a numerical field defining the parameter value.

N - Nominal; nominal filter parameters as discussed in Chapter 4.

IF - Simulation of an independent x- and y- channel MMAF, as discussed in Section 6.2.1.

NUM - Shifting of the filter initial conditions to alter the filter's numerical processing to establish performance sensitivity in the area, as discussed in Section 6.2.1.

C# - Non-ideal controller simulation. The numerical field identifies the value of $r(a)$, of Equation (6-1).

AR# - Aspect ratio sensitivity simulation; see Section 6.6.1.

S/N# - Signal to noise sensitivity simulation; see Section 6.4.

ZO# - Inertial initial z coordinate. The following numerical field identifies Z_{10} in kilometers.

NI - New image; redefined image to prevent FOV saturation; see Section 6.5.

RF - Filter retuned to improve performance for an off-design condition; see Section 6.5.

SA - Saturated target image; see Sections 6.5, 6.6.1, & 7.2.4.

BL* - Base line; Simulations with the "*" identifier have the "BL*" simulations for a baseline. Identifiers after the "*" in a designation code present variations from the baseline.

MD - Multiple dispersion hotspots; see Section 6.6.2.

IC - Inappropriate initial conditions applied to the filter to demonstrate robustness to handoff parameters at t_0 .

SPN - Increased separation of the hotspots in the target frame; see Section 6.6.2.

For example, following the designation code, the label;

T2G10MD*SPN

indicates a trajectory two, 10-g simulation with multiple dispersion hotspots, with a baseline simulation identified by *, with the variation on that baseline being an increased hotspot separation in the target image.

5.7 SUMMARY

This section provided the tools used in the simulation to evaluate the tracker performance. The designation code used for quick identification of the performance plots is also included.

VI. ALGORITHM INVESTIGATION

6.1 INTRODUCTION

This chapter describes the investigations conducted to evaluate algorithm characteristics, robustness, and sensitivity to parameter change. The analysis of these studies are also included in this chapter.

6.2 BIAS INVESTIGATION

Results of previous research encountered several unexpected biases and apparent rampings of the dynamics and centroid position error statistics [4]. To identify and correct these effects if appropriate, initially an indepth checkout and rewrite of the simulation software was conducted. Several software discrepancies were identified with at least one direct impact on simulation results. The rewriting and checkout of the software provided for an increased level of confidence in the simulation. The remaining unexplained results were investigated, assuming them to be a legitimate characteristics of the filter.

6.2.1 X-CHANNEL RAMPING

Research conducted by Loving [4] demonstrated a significant apparent ramping of the mean, plus or minus one standard deviation, in the x- dynamics errors and x-centroid errors committed by the filter. The maneuver initiation at $t = 2$ seconds generates a strong acceleration in the FLIR y- direction but only a minor deceleration along the FLIR x- direction, along which the velocity is predominantly directed. For this reason the substantially degraded performance in the FLIR x- direction was not anticipated. Figures A1 through A6 show the baseline x-channel propagated estimate $\hat{x}(t_i^-)$, (the minus indicating the estimate prior to measurement incorporation) for a five second simulation. The $\hat{x}(t_i^+)$ estimate (the plus indicating after measurement incorporation) results show identical trends. Qualitative analysis shows the ramp slope to be of decreasing magnitude with a decreasing maneuver turn rate. It was not possible to determine qualitatively if a slight ramping exists in the trajectory one five-second simulation performance plots (Figures A1).

Numerical processing can easily be the dominant error source in a standard Kalman filter implementation [5]. Since the filters are implemented in absolute coordinates (as discussed in Chapter 3), as the x- estimates increase in magnitude, the small difference of large numbers could cause

problems with numerical precision. In order to investigate this possibility, the initial conditions on the true and filter-assumed position states were changed. The new initial conditions force the large absolute coordinate processing to occur at the beginning of the simulation, where previously no significant ramping occurred. Figure A7 shows a trajectory two simulation with a 20-g sustained turn rate. As can be seen from this figure, no variation in the x- channel ramping is apparent as compared to Figure A5. It is concluded that numerical processing is not a significant contributor in the x- filter errors. This result is included here as additional validation to the simulation.

To define the apparent ramping better, the simulation was extended to ten seconds. Figures A8 through A19 present the x- channel extended simulation performance plots for trajectories 1 and 2, with both 10-g and 20-g cases for trajectory two. As is clear from the trajectory 1 plots (Figures A9 through 11), the ramping does not exist in the trajectory one simulation. From this result it is concluded that the ramping is a result of the sustained maneuver. This result is also implied by examining Figures A12 through A15 and A16 through A19, which qualitatively show the ramping beginning at $t=2$ seconds. Examination of the trajectory two, 20-g simulation (Figures A16-A19) shows a leveling off of the filter error beginning at approximately $t=7$ seconds. This implies a steady state error or maximum

overshoot has been reached, rather than an unbounded ramping process. Examination of the centroid error (Figure A19) shows the leveling off to occur at approximately $t=6.5$ seconds. Since the measurement (centroid position) is the sum of the positions due to dynamics, atmospheric effects and noise, the estimate of the centroid position must improve (at the direct measurement level) before improvements can be expected in the states x_A or x_D individually. This would be realized by a leveling off of the centroid error prior to that of the dynamic states. This, in fact, is the result indicated by Figures A17 and A19. Close study of the trajectory two, 10-g dynamics plots (Figures A13 and A14) does not provide conclusive results as to whether a steady state error bias has been reached. However, the centroid errors (Figure A15) does show leveling off beginning at approximately $t=8$ seconds. This implies that the dynamics position error is expected to reach steady state at approximately $t=8.5$ seconds based on the 20-g observations. Figures A14 and A15 support this rationale. It is observed that for the 10-g simulation the steady state error is less in magnitude and is reached at a later time than that for the 20-g case. Supporting the conclusion that the ramping is due to the sustained maneuver, are the trajectory three, 10-g simulation results (Figures A20 - A22). Figure A20 shows the x- dynamic errors beginning to ramp as the maneuver begins at $t=2$ seconds. At $t=3.5$

seconds, the maneuver ends and the error returns to approximately zero mean. This is confirmed by the centroid performance plot in Figure A21.

The mechanism largely responsible for this effect is linked closely to the MMAF tuning. When the maneuver begins at $t=2$ seconds, the y- FLIR component of the true target velocity undergoes large relative changes. The MMAF then adaptively "retunes" itself in an attempt to track the target. Specifically, elemental filter two (the wide FOV filter) attains a hypothesis conditional probability orders of magnitude above the other elemental filters. During this time, the true velocity change in the x- FLIR direction is rather benign. The x- FLIR trajectory is best matched by filter one's tuning. However, since the MMAF is driven to predominantly filter two by the filter two y- channel residuals, the MMAF is significantly mistuned for the x- FLIR channel. This mistuning results in poor state estimates, causing the increase in error. This effect is demonstrated by simulating the x- inertial flight path for a 20-g trajectory 2, while the y- inertial coordinate performs a trajectory one flight path. The results of this simulation are presented in Figures A23 through A26. A reduction in error in excess of 50% at $t=5$ seconds is realized due to the MMAF being relieved of the task of tracking a highly dynamic y- channel; allowing the MMAF to perform adaptive tuning specifically for the x- channel.

AD-A163 830 CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE ESTIMATION FOR TRACKING (U) AIR FORCE INST OF TECH

2/4

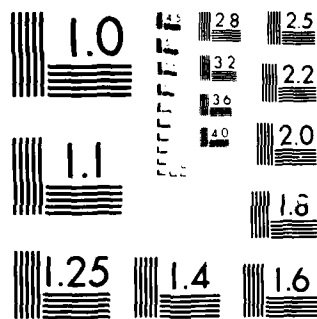
CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE
ESTIMATION FOR TRACKING. (U) AIR FORCE INST OF TECH

UNCLASSIFIED DEC 85 AFIT/GAE/ENG/85D-2

F/G 17/5

ML

[illegible]



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

The remaining error is conjectured to be a result of the inadequacy of a linear filter dynamics model for this application.

6.2.2 Y- CHANNEL RAMPING. Extending the simulation to ten seconds, as discussed in Section 6.2.1, showed a divergent ramping in the y position estimates for $t > 5$ seconds. This divergence is not due to the inertial maneuver, as can be seen in the trajectory one performance plots in Figures A27 through A30. Trajectory one shows a more severe divergence than is seen with trajectory two 10-g or 20-g cases; with the 20-g case having the least divergent trends. It is noted that in the absence of inertial maneuvering, the FLIR plane sees an accelerating target, due to FLIR/target geometry. The 20-g performance plots are presented in Figures A31 through A34. In all cases the filter has "knowledge" of the reduced performance, reflected in its computed error covariance matrix, as can be noted from the true RMS errors versus actual RMS error plots, Figures A27 and A31. The initial hypothesis about the cause of this effect, linked the divergence to the minimum range crossover point, which occurs at $t = 5$ seconds. However, simulation with a crossover point at 8 seconds showed no variation from the y- errors as seen in the previous figures. Examination of a nominal simulation greyscales plot time history identifies oscillations of the target image, and template,

in the y FLIR direction at various time intervals throughout the entire simulation. It is conjectured at this point that the divergence is due to an instability which grows slowly enough not to be significant until $t > 5$ seconds. The cause of this instability is conjectured to be model inadequacies encountered by using linear filter models for dynamics that would be better represented with a nonlinear model such as a constant turn-rate model [11,16]. It would be worthwhile to corroborate this conjecture by implementing a nonlinear dynamics model. This was not accomplished in this effort in order to perform higher priority analysis on the correlator /linear Kalman filter, before changing the structure of that algorithm.

6.3 NON-IDEAL CONTROLLER SIMULATION

The pointing controller is responsible for positioning the FLIR sensor in preparation for incoming measurements, and simultaneously pointing the line of sight along which the laser energy will be sent. The commanded movement $\hat{x}(t_{i+1}^-)_{\text{maf}}$, is ideally to be accomplished in one sample period, or less. In mechanical systems, inertial effects, servo lags, and backlash can cause significant time delays, resulting in pointing errors. This section investigates the results of the tracking algorithm with a first order lag pointing controller implemented as discussed

In Section 4.4.

6.3.1 Investigation Tracking performance was evaluated for a variety of controller time constants, a , in Equation (4-17). Trajectories one and two (with 10-g and 20-g turn rates) were investigated. In order to attach more physical meaning to the controller time constant, Equation (4-17) is rewritten as

$$\text{output}(t_i) = \delta_C(t_i) \cdot r(a) \quad (6-1)$$

where

$$\begin{aligned} \delta_C(t_i) &= \text{the commanded movement of the FLIR sensor} \\ &= [\hat{x}(t_i^-) - \hat{x}(t_{i-1}^-)]_{\text{maf}} \end{aligned}$$

$$\begin{aligned} r(a) &= \text{the fraction of the commanded movement that} \\ &\quad \text{the output will realize} \\ &= 1 - e^{-a\Delta t} \end{aligned}$$

$$\text{output}(t_i) = \text{the amount the FLIR sensor will actually move.}$$

For a fixed filter period, $r(a)$ indicates the percent of the commanded movement which can be accomplished in the sample period for the given system time constant. This study used a filter sample rate of 30 Hz.

The lower bounds for $r(a)$ was established at 0.90, with the corresponding $a = 69.08$, i.e., with lag time $1/a = 0.0145$ seconds. The system response modeled here is sufficiently slow as to provide x- pointing errors which cause the target to appear outside the FLIR FOV, within one

sample period. Values of $r(a)$ between 1.0 and 0.90 were then investigated. The test points are presented in Table 6.1

Table 6-1
Controller Test Points

a	1/a	$r(a)$	Trajectory 2 Turn Rate (g's)
		1.0	10
186.438	.0054	.998	10
102.534	.0098	.967	10
89.8719	.0111	.95	10
89.8719	.0111	.95	20
77.708	.0129	.925	10
77.708	.0129	.925	20
69.08	.0145	.90	-

Selected performance plots from the cases of Table 6-1 are presented in Appendix B. Performance plots which provide no additional insight are omitted in order to keep the analysis tractable.

Inspection of the $r(a) = .998$ simulation shows no significant variation in performance from the case without controller time lag effects, as can be seen in Figures B1-B7. Figures A1 through A15 present nominal performance for comparison. The next test point, $r(a) = .967$, demonstrates slight reduction in performance during the maneuver. Figures B8-B13 present the performance plots for this case. The increased error in the x- channel is evident in the increased slope of the error mean \pm one sigma (Figures B8-B10). The y- channel error increase is seen in the maximum overshoot at the onset of the maneuver, as well

as the ramping of the error for $t > 3$ seconds.

The ability to track the target prior to the maneuver with no apparent error bias or ramping is also evident from the figures. To investigate this robustness for $t \leq 2$ seconds, the greyscales of the target image and template are presented in Figures 6-1 and 6-2. Figure 6-1 presents the target image at frame one. The image is centered here since an ideal handoff from the acquisition phase to the tracking phase is assumed. The lack of a template image for this frame is noted, since there is no measurement prior to frame one. For subsequent frames the target image is offset due to imperfect pointing of the sensor, as is evident by Figure 6-2, corresponding to sample 40, at $t = 1.33$ seconds. Close study of the frames of data in this time interval provide valuable insight. First, the MMAF dynamic state estimates in both x- and y- FLIR directions are very good for $t \leq 2$ seconds. This indicates the pointing offsets are not being interpreted by the filter as a propagation error. This conclusion is further substantiated by observing the target image and template of the corresponding time interval. Figure 6-2 is the grayscale for the $r(a) = .95$ simulation, 40th frame, and is characteristic of all the frames of data for $t \leq 2$ seconds. As was previously noted, the target image is offset from the center of the FOV. Figure 6-2 shows that the template, formed by centering these offcentered images and performing temporal averaging, is clear, well defined,

RELATIVE PROBABILITIES = .9800, .0100, .0100

FRAME NUMBER = 1

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

-++0*0+-
+0*%XX*-
+0*%X*%*-
-+0*%*+-
---+-

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

Figure 6-1. Greyscale Target Image, Prior to Template Formation

RELATIVE PROBABILITIES = .9678, .0306, .0016
 FRAME NUMBER = 40

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      --+++-
12      -+*%X%0-
13      -0%$@%*-
14      -+*X$X0-
15      -+0*0-
16      --
17
18
19
20
21
22
23
24
  
```

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10      ----
11      -+****+
12      -+*X@%*-
13      -0%$@%*-
14      +0*%0-
15      -+++-
16
17
18
19
20
21
22
23
24
  
```

Figure 6-2. Greyscale Target Image and
 Template, Non Ideal Controller

and centered. This indicates that good estimates of apparent centroid position were supplied to the shifting operation, as defined in Equation (4-6). If poor apparent centroid estimates were made, the resulting templates would appear as smeared or stretched, which is clearly not the case in Figure 6-2. This template characteristic is described in detail in Section 6.5. Stated another way, with the target image significantly offset from the center of the FLIR FOV, in order to form consistently correct intensity functions (templates), the filter estimates of the target apparent centroid must be accurate. Examination of the propagated centroid errors show no significant error in the propagated estimates of the apparent target position, for all test points, $t \leq 2$ seconds. This leads to the conclusions that the filter propagated and updated estimates of the centroid are accurate. Since the dynamic state is estimated accurately for this time interval, the conclusion is drawn that the offset is interpreted by the filter as being due to atmospheric jitter, and that these states now estimate the sum of atmospheric and the pointing offsets. The filter's interpretation of the pointing offsets as errors in the atmospheric jitter states rather than as errors in the dynamics model provides robustness by maintaining nominal tracking performance in the presence of unknown pointing errors, for all test cases, in the interval $0 \leq t \leq 2$ seconds.

The 10- and 20-g trajectory 2 performance plots for $r(a) = .95$ (Figures B14-B26) show a different trend in the filter error for $t > 2$ seconds, as compared to results from the $r(a) = 0.967$ case. In both the 10 and 20 g, $r(a) = .95$ cases, the increasing x dynamics error takes a dramatic change shortly after the maneuver begins, that was not in the $r(a) = .967$ results. This large change in the filter errors is due to the target image moving significantly off the small FOV, resulting in the large FOV elemental filter abruptly assuming the highest $p_k(t_i)$ value. The different dynamic models in elemental filter one (small FOV) and elemental filter two (large FOV), causes the different state estimates apparent in the error shift. Inspection of the x- centroid plots for the 10- and 20-g $r(a) = .95$ case indicates a constant bias has been reached (Figures B19 & B25). Figure 6-3 shows the greyscale for $r(a) = .95$, frame 130. The greyscale resolution is insufficient to determine if the template is being smeared slightly. It is concluded that the filter is unable to obtain sufficiently precise state estimates to allow for performance comparable to that prior to the maneuver. No additional insights are gained from the $r(a) = .925$ case; for this reason the results are not presented.

RELATIVE PROBABILITIES = .2341, .4859, .2799
 FRAME NUMBER = 130

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

      ---
    -O**O-
  +O%$@X+
  **X@@X+-
  -O%$X*-
  -+++-
  
```

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

```

      ---
    -+O**+-
  ~+*X@$$-
  ~O%$@$$-
  ~+*X%0-
  -++++-
  
```

Figure 6-3. Greyscale Target Image and
 Template, Non Ideal Controller

6.4 SIGNAL-TO-NOISE RATIO SENSITIVITY ANALYSIS

In order to assess the filter's sensitivity to signal-to-noise ratio (S/N), a series of simulations were conducted with various values of S/N. The filter was not provided knowledge of the S/N change, since the measurement noise variance, R , is assumed constant in both the truth and filter models. Trajectory 2, with 10- and 20-g turn rates, was chosen to evaluate both maneuvering and non-maneuvering performance. The nominal three-hotspot target was used with parameters defined in Chapter 5. The various values of S/N were created by altering the maximum intensity of each of the hotspots. The signal to noise ratio is defined in this effort as the ratio of the maximum intensity, of each individual hotspot, I_{MAX} , with the FLIR measurement noise standard deviation, σ_n , as seen in Equation (3-15). Since $\sigma_n = 1.0$ for this effort, the S/N is simply the value of I_{MAX} . It is pointed out here that, since the intensities for the three hotspots are additive where the intensity profiles overlap, FLIR pixels with intensities greater than I_{MAX} are possible. For this reason, the results of this section should be used for trend information only, rather than an absolute indication of performance capabilities.

6.4.1 RESULTS. The performance plots for the S/N sensitivity analysis are presented in Appendix C. The results are rather uneventful, showing a gradual reduction

in performance as S/N is reduced. This reduction in performance is seen as an increase in the mean error and standard deviation in the x- and y- FLIR positions. The y-channel sees an increase in the maximum transient error due to the maneuver, and a steady state bias at the lower values of S/N. Time averaged error statistics can be useful for this analysis, and are presented in Tables 6-2 and 6-3.

Table 6-2
Time Averaged Statistics (mean \pm σ)
Error (Pixels) for .5 to 2.00 Seconds

S/N	\hat{x}^-	\hat{x}^+	\hat{y}^-	\hat{y}^+	g
20	.021 \pm .441	.014 \pm .392	-.017 \pm .390	.019 \pm .339	10
10	.015 \pm .452	.023 \pm .400	-.003 \pm .385	-.004 \pm .332	10
5	.056 \pm .482	.064 \pm .427	-.016 \pm .398	-.017 \pm .344	10
2	.156 \pm .714	.168 \pm .659	.015 \pm .544	.019 \pm .481	10
2	.156 \pm .714	.168 \pm .659	.015 \pm .544	.019 \pm .481	20
1.25	lost track				10

Table 6-3
Time Averaged Statistics (mean \pm σ)
Error (Pixels) for 3.5 to 5.0 Seconds

S/N	\hat{x}^-	\hat{x}^+	\hat{y}^-	\hat{y}^+	g
20	.301 \pm .454	.227 \pm .386	-.101 \pm .507	.040 \pm .461	10
10	.330 \pm .469	.255 \pm .404	-.105 \pm .485	.033 \pm .438	10
5	.395 \pm .502	.321 \pm .433	-.047 \pm .554	.090 \pm .505	10
2	.532 \pm .773	.462 \pm .714	-.148 \pm .693	-.005 \pm .579	10
2	1.198 \pm .779	.958 \pm .700	-.261 \pm .629	-.001 \pm .565	20
1.25	lost track				10

The decrease in performance as S/N is reduced, is evident from Tables 6-2 and 6-3. At S/N = 1.25 the filter lost track, which invalidated the time averaged statistics. The

reacquisition cycle allowed for continued tracking, providing the "low" S/N performance as seen in the performance plots, Figures C30 - C37. This results is consistent with previous research where the filter was shown to loose lock on the targets with S/N between 1.0 and 1.5, even in single hotspot scenarios [10,14,16].

As the S/N is reduced, the filter's ability to identify the target image from the background and FLIR measurement noises is impaired. This is evident in the filter tuning as seen in the filter-computed -vs- actual rms error performance plots. Figures C1, C14, C15, C21, C30, and C31 show that, as S/N is reduced, the actual RMS errors begin to increase above that of the filter- computed RMS errors. This indicates that the "filter is not looking hard enough at the measurements" [5] or, as in this case, cannot get enough information from the measurements, indicating the tuning is not optimal. The low S/N causes the true measurment variance to increase, creating the mistuned condition. Academically speaking, if one wanted to retune the filter for a reduced S/N, it would be necessary to increase the measurement error variance, R , in the elemental filters. Since this algorithm does not estimate R , the filter has no way to self diagnosis its inappropriate measurement error variance. The results herein demonstrate a moderate robustness for S/N equal to or greater than 2.

6.5 TARGET RANGE FROM SENSOR/PIXEL SIZE SENSITIVITY

ANALYSIS

The target range from the sensor can affect the filter's performance. Several issues come into play as range is reduced. First, FLIR plane velocities and accelerations increase. Filter performance may be reduced if the filter design point tuning is inadequate for the more severe FLIR plane dynamics. In addition to mistuning, the reduced range increases the size of the target image on the FLIR plane. As the target image increases in size, filling the FOV, the measurement update capability is reduced due to a partial loss of the target image.

Analogous to this effect is the issue of pixel size. By decreasing the size of each pixel in a fixed FOV array, the image size is increased, giving the impression the range has been reduced. The resultant dynamics appear less benign as the pixel size is decreased as well, like the case of actually reducing the range.

6.5.1 Investigation. In order to examine the MMAF's sensitivity to range, a series of simulations with trajectories that remain parallel to the $x_I - y_I$ plane were conducted, reducing the inertial z - coordinate, z_I from the nominal value of $z_I = 20,000$ meters. The filter was provided no knowledge or retuning to adapt to the parameter change. Trajectory one was used primarily, to remove

effects due to the maneuver, simplifying the tracking task. Trajectory two performance was also investigated to identify the algorithm's ability to track maneuvering targets at close range. In some simulations, an incorrect initial condition was introduced to the filter to study the filter's robustness to poor initial acquisition parameters. The filter's initial conditions are assumed to be provided by an acquisition phase, which is not within the scope of this effort. For simulation purposes, the filter's initial conditions on target position and velocity are obtained by projecting the respective true initial inertial vectors onto the FLIR image plane (assuming perfect handoff from the acquisition phase). The acceleration initial condition are obtained by differencing the true FLIR velocities over the first two sample periods, and dividing the difference by the sample period. The results are presented via performance plots, in Appendix D. Figures D1 - D6 shows the performance plots for $z_I = 15,000$ meters. The simulation was given the FLIR nominal acceleration initial conditions for $z_I = 20,000$ meters. The mismatched initial condition are evident in the x- error channels (Figures D1, D2, and D5). The incorrect initial acceleration propagates error down into the position states, as can be seen for $t < 1$ sec. The y acceleration initial condition does not introduce significant position error here, as the acceleration error is negligible. The results show a good recovery and nominal tracking,

indicating a robust filter for this off design condition. Filter robustness is further demonstrated by the $z_I = 10,000$ meters test point presented in Figures D7 - D12. The same acceleration initial conditions were provided to the filter as for the previous case. The initial transient here is more severe, as can be seen in the x channel plots (Figures D7, D8, and D10). This is due to the relatively larger error in the initial conditions.

The simulation was also conducted using trajectory 2, with a 10-g turn rate. As can be seen from the x- channel results, Figures D13, D14, and D17, the filter errors are very similar to the nominal case, with the exception of the initial transient due to the poor initial conditions. In the y- channel however, an approximately 70% larger error transient due to the maneuver commencing is observed. Additionally, a bias of approximately 0.5 pixels is noted, beginning after the transient (Figure D15). It is conjectured at this point that the reduction in performance is due to the mistuning at the off design point.

The filter's robustness at $z_I = 7,000$ meters is tested again by introducing the FLIR acceleration initial conditions for $z_I = 20,000$ meters. Here the x- acceleration error is a factor of 11.5 times that of the true initial inertial acceleration. Although the filter x- error transient is severe, the MMAF does recover by virtue of elemental filter two's dynamics and wide FOV. The results

for this simulation are presented in Figures D19 - D23. The trajectory 2, 10-g simulation for $z_I = 7,000$ meters was conducted with an improved initial estimate of the FLIR acceleration. The estimate was obtained from the change in the true velocity over the first two frames of data, as discussed previously. As can be seen from Figure D24 - D31, the improved initial accelerations aided the filter performance significantly, however the filter was not able to maintain a non-biased x- position error for $t < 2$ seconds (Figures D26, D27, and D30). Shortly after the start of the maneuver ($t=2.25$), an approximately 1.25 pixel shift in the mean error is observed in the x- dynamics (minus) estimate. This was caused by a sudden shift of the hypothesis conditional probabilities, $p_k(t_i)$, to weighting predominantly elemental filter two, which was maintaining a different vector of state estimates. A similar effect is again observed at approximately $t=3.9$ seconds, where the MMAF, which had once again predominantly weighted elemental filter one, shifts suddenly back to filter two. Figure D24 shows the x- true verses the filter rms errors plot. The plot shows the filter rms errors remain slightly greater than the true errors, implying an adequate choice of the filter's dynamic driving noise, Q_{DF} . In the y- channel, a 142% increase in the maximum overshoot, as compared to the nominal ($z_I=20000$ meter) case, due to the maneuver initiation is noted, followed by an approximately .66 pixel

bias in the y dynamic minus error estimate (Figure D28). Figure D25 presents the y- true versus the filter-computed rms errors. Here as in the x- channel, the plot indicates adequate \hat{Q}_{DF} tuning for the off-design condition.

A $z_I = 5,000$ meters, trajectory one, 10-g simulation was also conducted. Figure D32 shows the true versus filter-computed rms error plot for this case. As can be seen, true error grows rapidly, exceeding the filter rms error. This indicates the filter is inadequately tuned for the harsh target dynamics projected onto the the FLIR plane, at this range. At least one filter in the bank must be able to maintain lock on a target conducting the harshest (but reasonable) maneuver possible as seen at the minimum usable range, or else the bank of filters is deficient, and the MMAF cannot possibly maintain lock. The degraded performance indicated in Figure D32 is apparent in the x-position error plot of Figures D33 and D34. The filter lost track during this scenario.

The results for the 5,000 meters case raises an important issue. Is the poor tracking performance due to improper tuning, even possibly inadequate models; or is the filter not getting enough information from the measurement update because measurements are poor? Figure 6-4 shows a typical measurement during the simulation. As is observed, the image fills the small FOV's 8 x 8 array, not allowing the filter the entire target image for correlation. To

RELATIVE PROBABILITIES = .9800, .0100, .0100
 FRAME NUMBER = 3
 123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12 ----
13 -+000-- -+-
14 +*%*+--+0+
15 -+*%*0+--00+-
16 -+0++- -+-
17 -- -
18
19
20
21
22
23
24
123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11 -++++- ---
12 +0***+--+0+-
13 +*%*0++000-
14 -000++-+-
15 ---- -
16
17
18
19
20
21
22
23
24

Figure 6-4. Greyscale Target Image and
 Template, $z_0 = 5$ km

investigate this issue, a single hotspot simulation generating a small enough image to fit well in the small FOV, was conducted for the 5,000 meters case. Figure 6-5 shows the target image used for this simulation. Results of this case are presented in Figures D37 - D39. Examination of Figure D37 indicates an improvement in the x-measurements by the reduction in the true rms errors. Figures D38 and D39 show improved x- position estimates as a result of these improved measurements. The improved performance verifies the hypothesis that the target image size saturates the FOV, degrading the measurements. To investigate if filter tuning is an issue for this case, the single hotspot run was repeated with a retuned bank of filters. Each filter's dynamics noise variance, σ_{DF} , in the \underline{Q}_{DF} matrix defined in chapter 4, was increased by a factor of four, and filter one's time correlation constant for dynamics reduced from 3.5 to 2.5. The resultant performance is indicated in Figures D40 - D42. Figure D41 shows a decrease in tracking error relative to the previous case. The most substantial enhancement in performance is noted in the update estimate of position, Figure D42. This is expected since the higher filter dynamics model variance more accurately accounts for the larger amplitude maneuvers. A large increase in error is noted in the last second of the simulation. It is conjectured to be due to the limited adequacy in the "fixes" made in the last two cases. The

RELATIVE PROBABILITIES = .9800, .0100, .0100
 FRAME NUMBER = 2
 123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      ---
12      +%*0
13      -O$$$-
14      -*X$*-
15      +00+-
16      --
17
18
19
20
21
22
23
24
123456789012345678901234

```

```

1
2
3
4
5
6
7
8
9
10      --
11      +0*+-
12      -OXX*-
13      -*X$*-
14      -+***-
15      ---
16
17
18
19
20
21
22
23
24

```

Figure 6-5. Greyscale Target Image and
 Template, new Target Image,
 $z_0 = 5$ km

reduced target/sensor range seen in the later portions of the simulation results in more severe FLIR plane dynamics, as well as an increase in the size of the target image. It is noted that the minimum range crossover point is reached at $t = 5$ seconds. Figure 6-6 shows the single hotspot image at $t=4$ seconds. It is noted that the small FOV is filled. No additional insights were provided from the y - performance plots. They are not presented for this reason. It should be noted that the retuning of the MMAF is not presented as an optimal tuning. Although the true rms errors versus filter-computed rms errors indicate a well-tuned MMAF (Figure D40), it may be more appropriate to examine the true rms errors versus the filter-computed second central moment errors. This plot typically provides tuning insights where biases due to nonlinearities are present [6]. Figures D40 - D42 demonstrate that limited performance benefits can be realized by retuning. Insignificant performance variation was realized in a simulation in which the filter's dynamic model variance was increased to eight times the nominal value. However, as mentioned in Chapter 2, the addition of pseudonoise in the filter's dynamics model in an attempt to retune the elemental filters can result in a blurred distinction between the estimates and residuals based on different models. For this reason, it is conjectured that a limit on filter performance based on the elemental filters used in this effort, has been reached. It would be

RELATIVE PROBABILITIES = .0024, .9966, .0010
 FRAME NUMBER = 120

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      -
12      ----
13      -000+-
14      -+*X%*+
15      +0%***+
16      -OX%X%O-
17      -+*%*%*+
18      -+O++
19      ---
20
21
22
23
24
  
```

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10      ----
11      -+00+-
12      +0%*%*+
13      -+***XO-
14      -+***XO-
15      -+0%*%O-
16      -+00+-
17      ----
18
19
20
21
22
23
24
  
```

Figure 6-6. Greyscale Target Image and Template,
 new Target Image, $z_o = 5$ km

necessary to evaluate MMAF performance, with a dedicated elemental filter tuned specifically for a close range scenario with harsh dynamics, before statements on the adequacy of linear filter dynamics models can be addressed.

6.6 TARGET SHAPE INVESTIGATION

The purpose of this section is to identify filter tracking characteristics for various target image functions. This was accomplished by conducting simulations with specific target image shapes and qualitatively evaluating tracking performance. This study should not be considered a conclusive target shape sensitivity analysis, but rather an investigation of tracker characteristics and trends.

6.6.1 High Aspect Ratio Targets. An investigation into the effects of high aspect ratio target images on tracker performance was conducted. Aspect ratio and other target image parameters are defined in Section 3.5. Test points used for this study are found in Table 6-4.

Table 6-4
High Aspect Ratio Target Test Points

Case	Trajectory	AR	σ_{pvo}	# Hotspots	Hotspot Position (target frm)
nominal		1	1.414	3	$\pm 0.5e_{pv, lev}$
1	2/10 g	5	1	2	$\pm 2.5e_{pv}$
2	1	10	.707	2	$\pm 2.5e_{pv}$
3	1	.2	5	2	$\pm 2.5e_{pv}$
4	2/10 g	5	.55	2	$\pm 1.0e_{pv}$

Case one involved an elliptical target shape with the semi-major axis along the velocity vector. Figure 6-7 shows the target image greyscale at $t=5/30$ seconds and $t=150/30 = 5$ seconds. The inclination of the target image due to the maneuver is noted in Figure 6-7(b). The performance plots for this case are presented in Figures E1 - E6. The x-channel plots (Figures E1, E3, E4) show very poor tracking characteristics, with the filter losing lock slightly after the maneuver begins. The large error shift at $t=2.1$ seconds is the reacquisition cycle resetting the MMAF after the loss of track. Figures E5 and E6 show a lesser y-channel degradation in performance by a larger-than-nominal transient followed by a bias. The template and measurement greyscales were used to gain insight into the reduced performance. From Figure 6-7 it is noted that the target image extends beyond the small FOV array, in the x- FLIR direction. The measurement update obtained from this image creates poor state estimates, which in turn causes poor propagated state estimates. Case two was conducted to illustrate this effect. Figure 6-8 shows the initial centered target image and template. The "longer" target is used to induce larger errors to amplify the mechanism. With the target image filling the x- FOV, the filter's ability to correlate between the template and target image is impaired. The result is poor x- state estimates. These errors prevent proper image centering when forming the next template. The

RELATIVE PROBABILITIES = .9889, .0010, .0101
 FRAME NUMBER = 5

123456789012345678901234

1
 2
 3 (A)
 4
 5
 6
 7
 8
 9
 10
 11 ---+-----
 12 -++O***XXXX**OO+---
 13 ---+OO***XXXX**O+---
 14 ---+-----
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24

FRAME NUMBER = 150

123456789012345678901234

1
 2
 3 (B)
 4
 5
 6
 7
 8
 9 --
 10 ---+---
 11 ---+O***O+-
 12 -+O***X**+-
 13 -+O***X***O-
 14 ---+O***O+-
 15 ---+OOO++
 16 ----
 17
 18
 19
 20
 21
 22
 23
 24

Figure 6-7. Greyscale Target Images
 High AR case 1

123456789012345678901234

15
16
17
18
19
20
21
22
23
24

```

1
2
3
4
5
6
7
8
9
10
11          -----
12  --+O***XX#####%*O--
13  -+OXXXXX%#####XX%O--
14          -----++-----

```

15
16
17
18
19
20
21
22
23
24

-107-

temporal averaging of non-centered target images causes the template to be stretched or smeared. This is evident in Figures 6-9 through 6-11, which show the simulation results from case two. The Figure 6-9 greyscale, ($t=.5$ seconds), shows the target image walking off the FLIR image plane while the template maximum intensity has been stretched toward the right. The deformed template prevents the algorithm from efficiently correlating the target image and the template, impairing the filter from improving its estimates. In addition, it was observed that, if the algorithm can achieve some stability (but bias as well) in the tracking of the target image on the FLIR plane, the template will form offcentered, corresponding to this image. This mechanism can be seen in Figures 6-10, $t=1$ second, and Figure 6-11, $t=2$ seconds. The resultant performance is seen in the steady state x- errors, presented in Figures E3 and E4.

The counterpart to case two was simulated in case three. Here an ellipse with the semi-major axis in the y FLIR direction is used. Examination of the greyscales showed that again the image, "walked off" from the center of the FOV, with its offcentered template. Figures E7 and E9 show nominal x- channel tracking for trajectory one. The y- channel position errors shown in Figure E10 indicates substantially degraded performance. It is noted from the true rms versus filter-computed rms error plot, (Figure E8),

RELATIVE PROBABILITIES = .9907, .0010, .0083
 FRAME NUMBER = 15
 123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11
12      ----++000**%*%*%*%*%*
13      --++0*%X$#@#####
14      -----+-+~+---+++
15
16
17
18
19
20
21
22
23
24
  
```

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      -----
12      ----++00**%XXX$###@###
13      -+++++0000*%XX#@#####
14      -----
15
16
17
18
19
20
21
22
23
24
  
```

Figure 6-9. Greyscale Target Image and Template, High AR case 2

RELATIVE PROBABILITIES = .9894, .0010, .0096
 FRAME NUMBER = 30
 123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11
12      ---++000**%*%*%*%*%*
13      --++0**%X*%*%*%*%*%*%
14      -----+0000000000
15
16
17
18
19
20
21
22
23
24
  
```

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      --
12      --++00**%XXX*%*%*%*%*%
13      -  -++0000**%XX*%*%*%*%*%
14      -----
15
16
17
18
19
20
21
22
23
24
  
```

Figure 6-10. Greyscale Target Image and Template, High AR case 2

RELATIVE PROBABILITIES = .9260, .0010, .0730
FRAME NUMBER = 60

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

-----+++0000000
---++0***X#####
- --000***% % % % %

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

--++00***X*#####
--++00***XX*#####

Figure 6-11. Greyscale Target Image and
Template, High AR case 2

that the filter was not aware of its degraded performance, which was not the case in the previous test case. As a result, the reacquisition cycle was not used and the steady state error not realized until late in the simulation as seen in the y- position error plot in Figure E10.

To assess the filter's ability to correlate the high aspect ratio target shape function without the penalty due to overextending in the FLIR plane, case four was conducted; case four utilizes a target image that fits "well" into the small FOV. Figure 6-12 shows the target image and template for t=5 seconds. Again note the inclination in the image due to the maneuver. Figures E11 - E16 show the performance plots for this case. A degradation in performance is noted in the y- channel as a larger transient due to the maneuver. Both the x- and y- channels exhibit a bias after the beginning of the maneuver. However, case four shows a significant improved performance as compared to case one. This again identifies the effects of too large a target image or too small a FLIR tracking window on the state estimates. A slightly reduced ability to track high aspect ratio targets is noted when image extension beyond the FOV boundaries is not a factor.

6.6.2 Multiple Dispersion Hotspots. This section investigates tracker performance against a multiple hotspot, multiple dispersion target image. The dispersion of a hotspot image defines the spatial spread or gradient of the

RELATIVE PROBABILITIES = .9646, .0150, .0205
 FRAME NUMBER = 150

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      -+++-
12      -+*XX*+
13      +*%O+
14      ----+-
15
16
17
18
19
20
21
22
23
24

```

123456789012345678901234

```

1
2
3
4
5
6
7
8
9
10
11      ---
12      -O*%O-
13      +O%X%O+
14      -+++-
15
16
17
18
19
20
21
22
23
24

```

Figure 6-12. Greyscale Target Image and Template, High AR case 4

hotspot intensity; analogous to the covariance, in a bivariate Gaussian probability density function. For this study, a two-hotspot target model, each spot having a different dispersion matrix, was used. The effect on the target image is to create a more distinctive target shape by incorporating a steep intensity gradient hotspot, with a nominal hotspot. To this end, the following 2-dimension target frame dispersion matrices were used:

$$\underline{P}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} ; \quad \underline{P}_2 = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix}$$

where in terms of the notation of Section 3.4,

$$\underline{P}_i = \begin{bmatrix} \sigma_{vi}^2 & 0 \\ 0 & \sigma_{pvi}^2 \end{bmatrix} \quad i=1,2$$

and \underline{P}_1 is the nominal dispersion matrix \underline{P} . Note both hotspots have circular constant-intensity contours.

Tables 6-5 and 6-6 present the time averaged statistics for a trajectory 2, 10-g simulation for this case. The first entry is a two-hotspot nominal dispersion baseline. The second presents the multiple dispersion hotspot results.

Table 6-5 Trajectory 2, 10 g; Two Hotspots
Error (Pixels) from Time .5 to 2 Seconds

Dispersion	\hat{x}^-	\hat{x}^+	\hat{y}^-	\hat{y}^+
$\underline{P}_1 = \underline{P}_2$	-.014 \pm .444	-.002 \pm .389	-.026 \pm .385	-.028 \pm .332
$\underline{P}_1 = 10\underline{P}_2$.056 \pm .449	.063 \pm .395	.014 \pm .388	.013 \pm .336

Table 6-6 Trajectory, 10 g; Two Hotspots
Error (Pixels) from Time 3.5 to 5 Seconds

Dispersion	\hat{x}^-	\hat{x}^+	\hat{y}^-	\hat{y}^+
$P_1 = P_2$.307 \pm .464	.232 \pm .401	-.455 \pm .425	-.321 \pm .368
$P_1 = 10P_2$.455 \pm .466	.383 \pm .397	-.460 \pm .466	-.311 \pm .366

From Tables 6-5 and 6-6, it is apparent that although a decrease in performance is perceptible for the multiple dispersion case, the degradation is insignificant. This result is confirmed by comparison of the performance plots in Figures E17 - E20, (two hotspots with $P_1 = P_2$), and Figures E21 - E24 ($P_1=10P_2$). The result implies the algorithm is capable of correlating the more distinct, complex shape function encountered here, with little difficulty, resulting in insignificant performance degradation.

In order to define each hotspot more clearly on the FLIR image plane, the hotspot separation was increased by a factor of 5. This reduces the overlapping of the individual hotspot intensity functions, causing a more distinct, complex shape functions. The performance of the tracker against this target showed significantly larger errors, in the form of an immediate ramping of the errors for $t > 0$. This result is presented in the performance plots of Figures E25 - E28. This result is not surprising. As the hotspots

become more distinct, changes in the target shape functions due to changing target/sensor geometry are more distinguishable. This results in correlation problems with the slower changing template, resulting in increased state estimate errors. This effect was also observed in simulations in which a single dispersion matrix was used for both hotspots, where hotspot separation caused a distinct, complex, changing shape function.

6.6.3 Target - Decoy Experiment. For many years targets have employed flares or chaff, as decoys to defeat infra-red sensors. This section presents an experiment in a sudden dramatic change of the target image, to simulate such a flare or chaff release. It should be note that this section is not a tactical decoy sensitivity analysis, but an experiment to provide insight into tracker characteristics.

The experiment consists of tracking a single hotspot target for one second. The target is performing a "trajectory one" flight path. Figure 6-13 shows the target image and template during this time. At $t=1$ sec., a second hotspot is turned on, "behind" the centered target image. This simulates the release and subsequent ignition of a flare or chaff. The second hotspot is identical to the first, except that the maximum intensity is a factor of 2 greater. Once the second hotspot is turned on, the 2 hotspots are separated in the FLIR y direction to simulate the target movement away from the chaff. Figure 6-14 shows

RELATIVE PROBABILITIES = .9800, .0100, .0100
 FRAME NUMBER = 2
 IMAX = 0.000

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11 +-
12 0%0
13 +*0-
14 -
15
16
17
18
19
20
21
22
23
24

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11 --
12 -0*-
13 -***+
14 +-
15
16
17
18
19
20
21
22
23
24

Figure 6-13. Greyscale Target Image and
 Template, Target/Decoy Experiment

the target image and template at $t=1$ second. Figures 6-15 through 6-18 present sequential target images and templates, showing the tracker's reaction to the changed shape function. As the greyscale presents, the tracker immediately shifts off the target toward the centroid of the two hotspots (Figure 6-16). Template realization of the second hotspot, which now corresponds to the target, is slow due to temporally averaging essentially zero intensity in the corresponding pixels prior to the chaff release. This is compounded by the target/decoy relative dynamics, shifting the target's image toward the edge of the FOV as the algorithm predominantly tracks the decoy (Figure 6-15). As the state estimate errors increase, the template's predominate image becomes smeared due to inaccurate centering of the new measurement image (Figures 6-17 and 6-18). The tracker ultimately locks onto the decoy, allowing the target to move off the FOV. Figure 6-18 shows the results for $t=2.0$ seconds; the tracker is locked on the decoy, with an appropriate template. The Monte Carlo performance plots of the experiment are presented in Figures E29 through E34. Figures E31 - E34 show the immediate divergence of the position errors at the release of the decoy. The apparent x- channel recovery shown in Figures E31 and E32 is misleading. The reduction in error is a result of the target/decoy mechanism employed in the simulation. The target has moved "underneath" the decoy on

RELATIVE PROBABILITIES = .9966, .0010, .0024

FRAME NUMBER = 31

IMAX = 60.000

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

+*0 +-
0%0 -s#*
--- +@#%
 **+

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

--
-0*-
-***+
++

Figure 6-14. Greyscale Target Image and
Template, Target/Decoy Experiment

RELATIVE PROBABILITIES = .0395, .0010, .9595
 FRAME NUMBER = 33
 IMAX = 60.000

123456789012345678901234

1		
2		
3		
4		
5		
6		
7		
8		
9		***-
10	-	*##0
11	+++	000+
12	0%0	-0+
13	--+	

14
15
16
17
18
19
20
21
22
23
24

123456789012345678901234

1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11	--	
12	-00+--	
13	-0*+---	
14	---	

15
16
17
18
19
20
21
22
23
24

Figure 6-15. Greyscale Target Image and
 Template, Target/Decoy Experiment

RELATIVE PROBABILITIES = .2334, .7100, .0566
 FRAME NUMBER = 36
 IMAX = 60.000

123456789012345678901234

1	
2	
3	
4	
5	
6	
7	
8	
9	+-
10	0\$X+
11	*#00
12	+%%-
13	--
14	
15	-0+
16	+X*-
17	-00
18	-
19	
20	
21	
22	
23	
24	

123456789012345678901234

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	--++-
12	+++00-
13	+0+++-
14	-
15	-
16	
17	
18	
19	
20	
21	
22	
23	
24	

Figure 6-16. Greyscale Target Image and
 Template, Target/Decoy Experiment
 -121-

RELATIVE PROBABILITIES = .0165, .1855, .7979
FRAME NUMBER = 45

IMAX = 60.000

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

-
- * % +
+ @ # %
- X @ 0
- +

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

++-
-0*+
+*+*+
--+00-
-- ---

Figure 6-17. Greyscale Target Image and
Template, Target/Decoy Experiment

RELATIVE PROBABILITIES = .0010, .4472, .9518
 FRAME NUMBER = 60
 IMAX = 60.00

123456789012345678901234

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
123456789012345678901234

-+O+
 +\$#*-
 +\$#%-
 -O*+

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

-+-
 -%\$*
 +\$#%-
 -*XO
 --

Figure 6-18. Greyscale Target Image and Template, Target/Decoy Experiment

the FLIR plane, resulting in the reduction of x- channel errors. The error reduction was in no way achieved by the tracker. Figures E29 and E30 show the x and y true verses actual rms errors. These plots indicate the filter is unaware it is tracking a decoy.

6.7 SUMMARY

This chapter has presented the investigations and analysis conducted to evaluate the three-filter Bayesian MMAF tracker algorithm. A summary of the results along with the conclusions drawn from the analysis of this chapter, are presented in Chapter 7.

VII. CONCLUSIONS AND RECOMMENDATIONS

7.1 INTRODUCTION

This chapter summarizes the observations and conclusions drawn in Chapter 6. Insights from each investigation are combined to form conclusions of the overall algorithm characteristics. Recommendations for tracker evolution and further investigation are included.

7.2 CONCLUSIONS

7.2.1 Bias Investigation. The x- position bias errors presented in Figures A3 and A5 are a result of the algorithm tracking a target performing a high-g, constant turn rate maneuver. The apparent ramping is, in fact, an error transient, prior to reaching a steady state bias. The simulated maneuver is more severe than would exist in an anticipated tracking scenario. The step input of 10-g or 20-g accelerations places a high demand on the filter, while a reduced transient would be expected with a more realistic maneuver. A major cause of the bias is the MMAF mistuning the x- channel. In order to maintain lock on the highly dynamic y- channel transient, the MMAF adaptively tunes to predominantly elemental filter two, i.e., the wide FOV filter tuned for harsh maneuvers. This MMAF tuning is

not suited for the relatively benign x- channel dynamics, resulting in reduced tracking performance. Results show that over 50% of the bias in the 20-g trajectory two scenario, is due to this mistuning. This motivates the concept of individual x, y channel, multiple model adaptive filtering, allowing adaptive filtering for the x and y FLIR channels independently. However, this approach would be computationally more burdensome, requiring twice the residual monitoring. The remaining bias is attributed to the inadequacy of the linear dynamics model for very harsh target dynamics. This motivates the consideration of an extended Kalman filter, to allow for the nonlinear, constant turn rate dynamics model [7], while still maintaining the linear measurement update formulation via the enhanced correlator.

7.2.2 Non-Ideal Controller. Investigation of filter performance with first order lag pointing errors showed the algorithm to be robust to dynamic lags, within the limitations of the modeled FOV. This robustness is attributed to the atmospheric states absorbing the pointing errors, allowing the dynamics states to ignore the controller error, for relatively benign target trajectories. A moderate degradation in performance was noted for highly dynamic targets. However, the tracker was able to maintain lock on a 20-g maneuvering target. The limiting factor in the study was the size of the FOV. With sufficiently slow

controller dynamics, the target image is not maintained in the small FOV. For the FOV used in this effort, the filter immediately lost track when $r(a) \leq .90$, where $r(a)$ is defined in Equation (6-1). This motivates investigation of a larger FOV for an elemental filter tuned for benign dynamics. In addition, it would imply that a pointing reacquisition cycle would be advantageous, to bring back on line filters which have lost track due to pointing errors, and recenter the target image on the FOV. It is noted that the reacquisition cycle discussed in Chapter 3 assumes the FLIR sensor is instantaneously pointed at the new state estimate. This assumption is not valid for the non-ideal controller. To compensate for the controller lags in the reacquisition cycle, it may be possible to adjust the commanded control by the amount $1/r(a)$. By purposely commanding $[1/r(a)] \delta_C$, the actual control (the output as seen in Equation (6-1)) is changed to equal to the desired control, δ_C . This compensation would require some type of estimate of the controller lags. It is reasoned that, even without a FOV size limitation, a sufficiently slow system would cause filter divergence. This would be as a result of the pointing errors becoming so large as to prevent the atmospheric states dynamics model from propagating adequate estimates.

Although tracker performance did show a certain amount of robustness, filter knowledge of the controller dynamics

on performance specifications and controller system dynamics. Increasing the measurement noise variance may provide an ad-hoc method to compensate for the controller errors, without requiring additional computation. The pointing errors term, as seen in Equation (4-18), could be incorporated into the measurement noise, $v(t_i)$, allowing the measurement equation to be written in the form of Equations (4-10) and (4-13).

7.2.3 Signal to Noise Ratio Sensitivity. The S/N sensitivity analysis showed a smooth decrease in performance with decreasing S/N, from $S/N = 20$ to $S/N = 2$. By reducing the S/N a factor of 10, the mean error increased less than 1.0 pixel in the FLIR x- and 0.2 pixels in the FLIR y coordinate, during a steady state, sustained 20-g maneuver (with initial velocity mostly in the x- direction and the maneuver acceleration predominantly in the y- direction). Equally as important, the maximum standard deviation increase was approximately .33 pixels and .1 pixels in the FLIR x and y coordinates, respectively, under the same conditions. The standard deviation is significant as it is necessary to deposit sufficient laser energy within a specific region on the target to achieve a reasonable probability of kill, and not "paint" the target. An inherent weakness in the enhanced correlator/linear Kalman filter is the larger nominal tracking error standard deviation relative to the extended Kalman filter [4,16]. Low

standard deviation magnitudes and sensitivity are essential if this algorithm is to remain a viable alternative tracking algorithm.

7.2.4 Target Range from Sensor. Robustness analysis showed that the MMAF maintained track, for handoff initial accelerations a factor of 11.5 in error, at ranges of 8,600 meters, with nominal tuning as described in Section 5.5. This result implies that a highly accurate velocity or acceleration acquisition handoff is not necessary. This softens the requirements on the acquisition algorithm.

Performance analysis of tracking targets at various inertial crossing distances demonstrated a gradual reduction in performance as z_1 was reduced from 20 km to 7 km, for both benign and 10-g maneuvers. At 5 km crossing range the algorithm lost track. The mechanism behind the reduced performance and eventual loss of track is two-fold. First, the increased size of the target at close range, saturated the small FOV. The saturated FOV prevents adequate correlation between the target image and the template. This result is not surprising. The sizing of the FOV must take into account the target image size relative to the FOV. A second, and more pressing result is the inadequacy of the elemental filter dynamics model used in this effort for close tracking scenarios. The limited performance benefit gained by retuning the elemental filters indicates a performance limit may have been reached. The addition of

more pseudo-noise in an attempt to improve tuning is not considered a viable approach, for the reasons presented in Chapter 2. This motivates the addition of another linear elemental filter, specifically tuned for the extremely harsh maneuvers encountered in the close tracking scenarios. However, in light of the retuning attempts made in this effort, it may be necessary to include a nonlinear dynamics model in the MMAF algorithm, in order to improve performance in close range tracking scenarios.

7.2.5 Pixel Size. Analogous to range from sensor, is the issue of pixel size. Results of this effort clearly establish the degraded performance associated with saturation of the FOV. To avoid this with a fixed number of pixels, it is necessary to size each pixel large enough to provide good image shape for correlation. An additional benefit in an increased pixel size is the resulting FLIR plane dynamics are more benign. On the other hand, larger pixels have reduced resolution, since the intensity is averaged over the larger pixel area. This results in reduced performance, due to poorer correlation. This is evident in the significantly reduced performance realized when the target image extends outside the center 8x8 pixel array into the outer frame, so that filter performance becomes more dependent on an elemental filter with a large FOV composed of pixels with poorer resolution.

7.2.6 Target Shape Investigation. Results from the high aspect ratio simulations show a slight degradation in performance for tracking target images with aspect ratios as high as 5, while the target image does not saturate the small FOV. The degraded performance is realized in the coordinate along the semi-major axis of the ellipse and is a result of less accurate correlation along that direction. The performance degradation is severe if the semi-major axis extends over the entire FOV, saturating a FLIR coordinate direction. The resultant poor state estimates cause a smearing or stretching of the template due to the non-centering of successive frames of data, in the formulation of the template. A steady state bias is formed in many cases when sufficient stability of the tracker allows time for the stretched, offcentered template to reform, emulating the apparent target image. If the ellipticity is sufficient to prevent saturation in the semi-minor axis FLIR plane direction, the corresponding channel will produce nominal state estimates, provided that the saturated channel does not "walk" the image off the FOV. The template smearing or stretching mechanism was also observed in non-elliptical target image simulations. Poor target centroid position state estimates as a result of target dynamics or rapidly changing shape functions are the prerequisites for this effect.

It is desirable to define the result "FOV saturation".

Simulations in which the center 8 x 8 "small FOV" was fully populated with pixel intensities, providing no edge to the shape function, clearly displayed degraded performance. However, it must be noted that the center 8 x 8 array is padded with the corresponding data and noise, extending the 8 x 8 array to 24 x 24 pixels for FFT generation, for the small FOV filters. However, the frame surrounding the center 8x8 array is composed of pixels with significantly lower resolution as compared to the center array. It should be noted here, that the poorer resolution in the padding pixels are a result of the FLIR sensor modeling in the truth model. This should not be confused with the reduced resolution of the larger pixels, used in the wide FOV elemental filter. Results show that performance is significantly reduced where target/template correlation involves the outer frame pixels. As a result, the FLIR saturation is defined as the condition where the FOV is unable to display well defined, perceptible edges in the target image. For the FOV used in this effort, this requires the target image to be well within the center 8x8 "small" FOV array for elemental filter one or three tracking.

Simulation results also provide insights into more complex shape functions. The results from multiple dispersion hotspots showed insignificant performance degradation as compared to the two hotspot baseline case.

This is significant since actual multiple hotspot target intensities functions rarely have similar gradients surrounding all intensity peaks. A significant performance degradation was noted when the individual hotspots were separated sufficiently to define individual intensity gradients better. This is due to the dramatic target image changes that are realized as the target/sensor geometry changes. With the larger hotspot separations, even the same slow angular rates can yield more dramatic changes in the image. The template, with its temporal smoothing formation, reacts too slowly to allow adequate target image/template correlation. Thus, with larger hotspot separations, the temporal averaging of a target undergoing angular orientation changes causes greater impairment of a well resolved target image template that might well match the target shape without the distortions caused by averaging. This result was observed in multiple and singular dispersion simulations. Adjusting the smoothing constant to allow for faster template dynamics should be done carefully. Discounting previous measurements prematurely defeats the noise rejection function of the smoothing algorithm. This result should be taken into account when establishing pixel resolution, and a tracking range design point. Too much detail, with sufficient separation, can result in a target image with a highly dynamic shape function, impairing correlation. This could establish a minimum tracking range

in addition to model adequacy discussed in Section 7.2.4.

7.2.7 Target-Decoy Experiment. This set of simulation results displays the tracker's indifference to the target in order to track the hotter decoy. This implies the correlation algorithm "prefers to fit" the current template into the hotter of the two images. As the template begins to acknowledge the second hotspot, the MMAF appears to track the centroid of the two images. However, the target/decoy dynamics cause the hotspots to separate, preventing the inclusion of both in the FOV. At this point, the MMAF rejects the target image for the decoy. It is noted that the template "behaved" as desired, rejecting for the most part the formation of a second hotspot, maintaining an appropriate shape function estimate. The "weak link" in rejecting the decoy is clearly the correlator. This result is disappointing. It had been hoped that, after tracking the target successfully for 30 frames, the filter would reject the second hotspot as noise because of its different dynamics and intensity shape function. This experiment indicates a viable means of defeating the tracker. Investigation into this problem is warranted. An additional problem scenario which is implied by this result is that of tracking multiple targets which would co-occupy the same FOV. These results motivate a smaller FOV. The inability of the algorithm to reject a large hotspot implies the need for isolating the target image as much as possible. In

addition, it may be possible to reject the decoy by the use of artificial intelligence concepts. Information supplied to the algorithm, based on a study of the residuals and image characteristics, could provide a means of logically and consistently rejecting chaff or other countermeasures.

7.3 RECOMMENDATIONS

The following recommendations are made as a result of this effort.

1. Assess performance tradeoffs for individual x, y channel multiple model adaptive filtering.
2. Assess the performance tradeoff for implementing an extended Kalman filter with a nonlinear dynamics model (constant turn rate) [7] and linear measurements. This formulation would not require the high computational loading associated with nonlinear measurements [10,14].
3. Perform a close range tracking scenario performance tradeoff. This analysis should include:
 1. benefits of nonlinear constant turn rate dynamics model

- ii. optimal tuning of the current algorithm to evaluate if Q_{DF} scheduling as a function of range is feasible. This could be accomplished with sparse active sensor data to supply range and possibly range rate, to identify the appropriate Q_{DF} .
 - iii. the addition of an elemental filter tuned specifically for the close range scenario with harshest expected target maneuvers as seen at that range
 - iv. investigate alternative trackers such as the extended Kalman filter formulation [16]
 - v. effect of multiple hotspot definition and separation at close ranges
4. Conduct an indepth target/decoy sensitivity analysis. Tracker sensitivity to the following parameters should be investigated:
- i. decoy intensity, shape, and size
 - ii. separation from target during decoy ignition
 - iii. tracking time prior to decoy release

iii. tracking time prior to decoy release

iv. analysis of image and/or residual characteristics that might be useful to distinguish between true targets and decoys, perhaps by invoking A.I. concepts.

5. Incorporate an additional filter, with a wide FOV and benign dynamics model. This filter could be used as an acquisition filter as well as aid in tracking performance.

APPENDIX A
BIAS INVESTIGATION

The figures contained in the following appendices
(A through E) use the designation code described in
Section 5.6

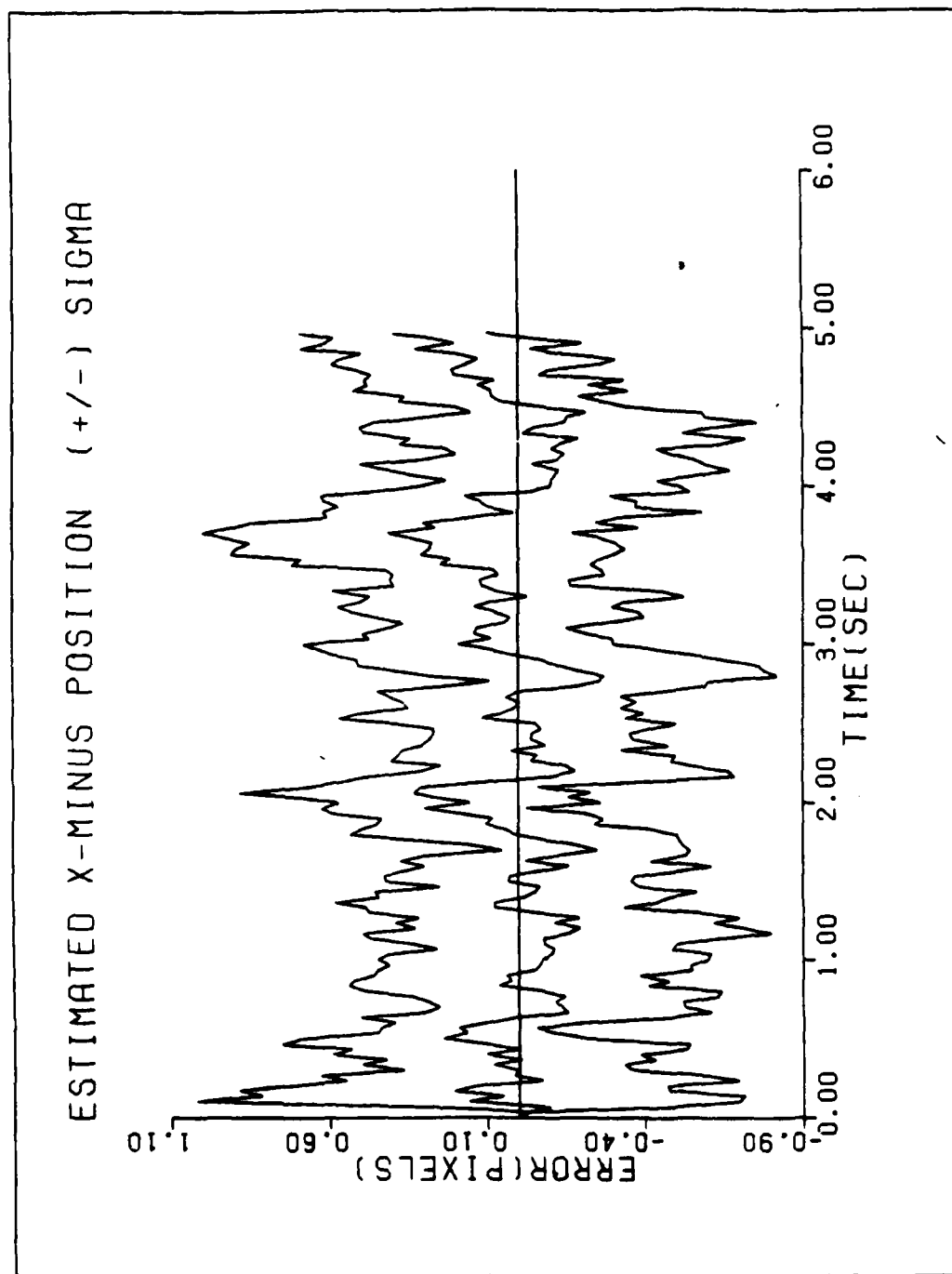


Figure A1 Performance plot for T1N

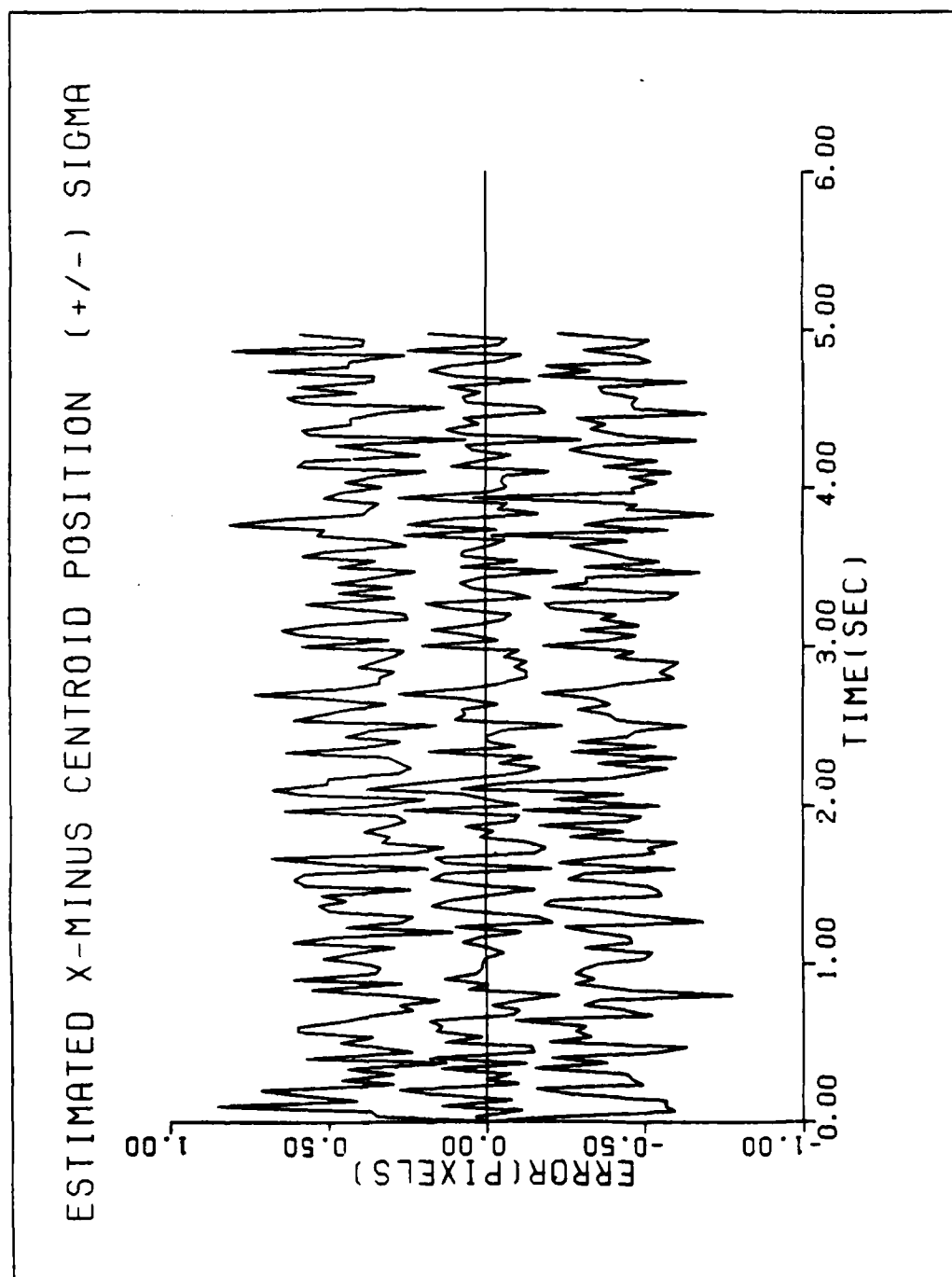


Figure A2 Performance plot for T1N

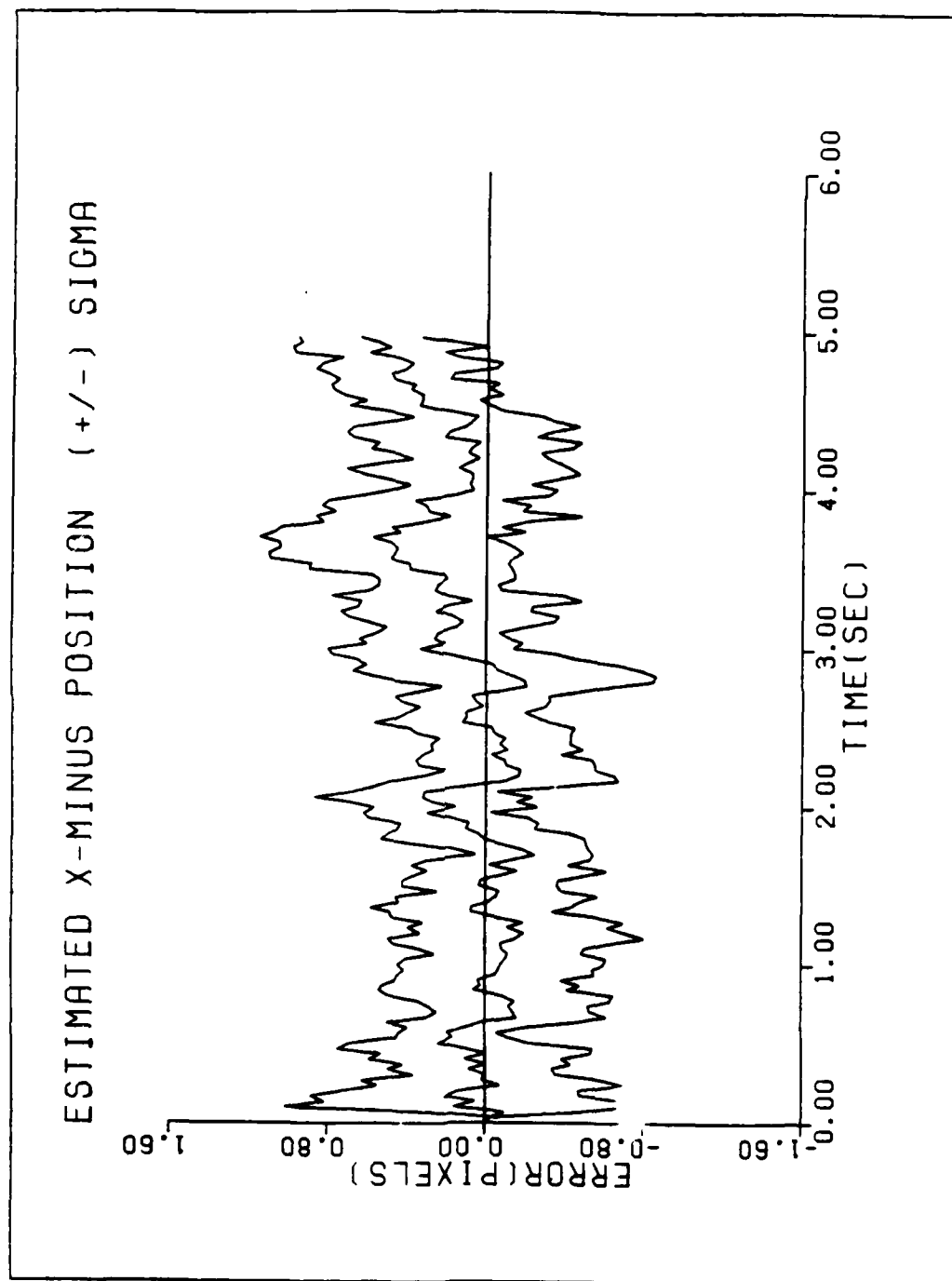


Figure A3 Performance plot for T2G10N

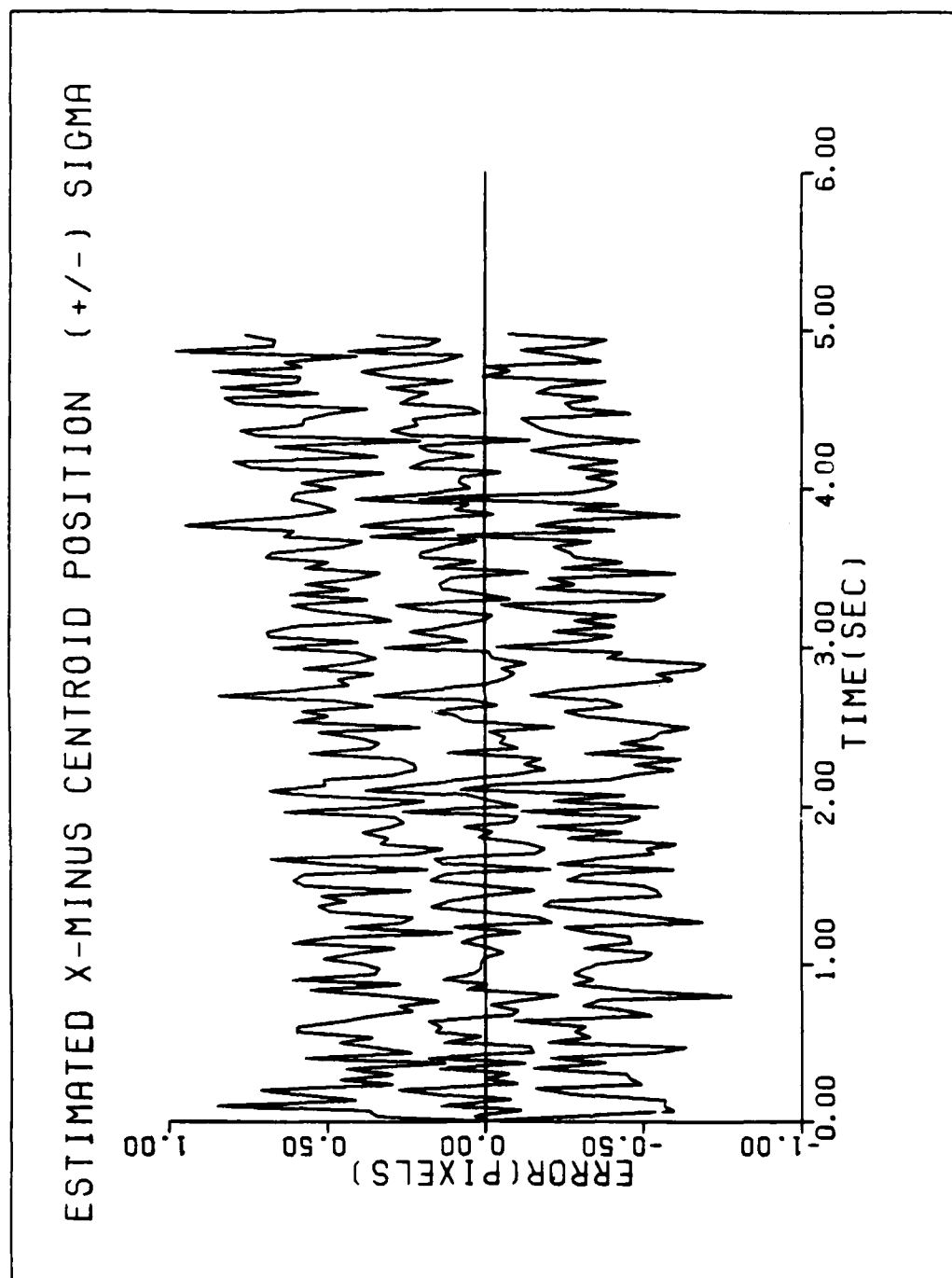


Figure A4 Performance plot for T2G10N

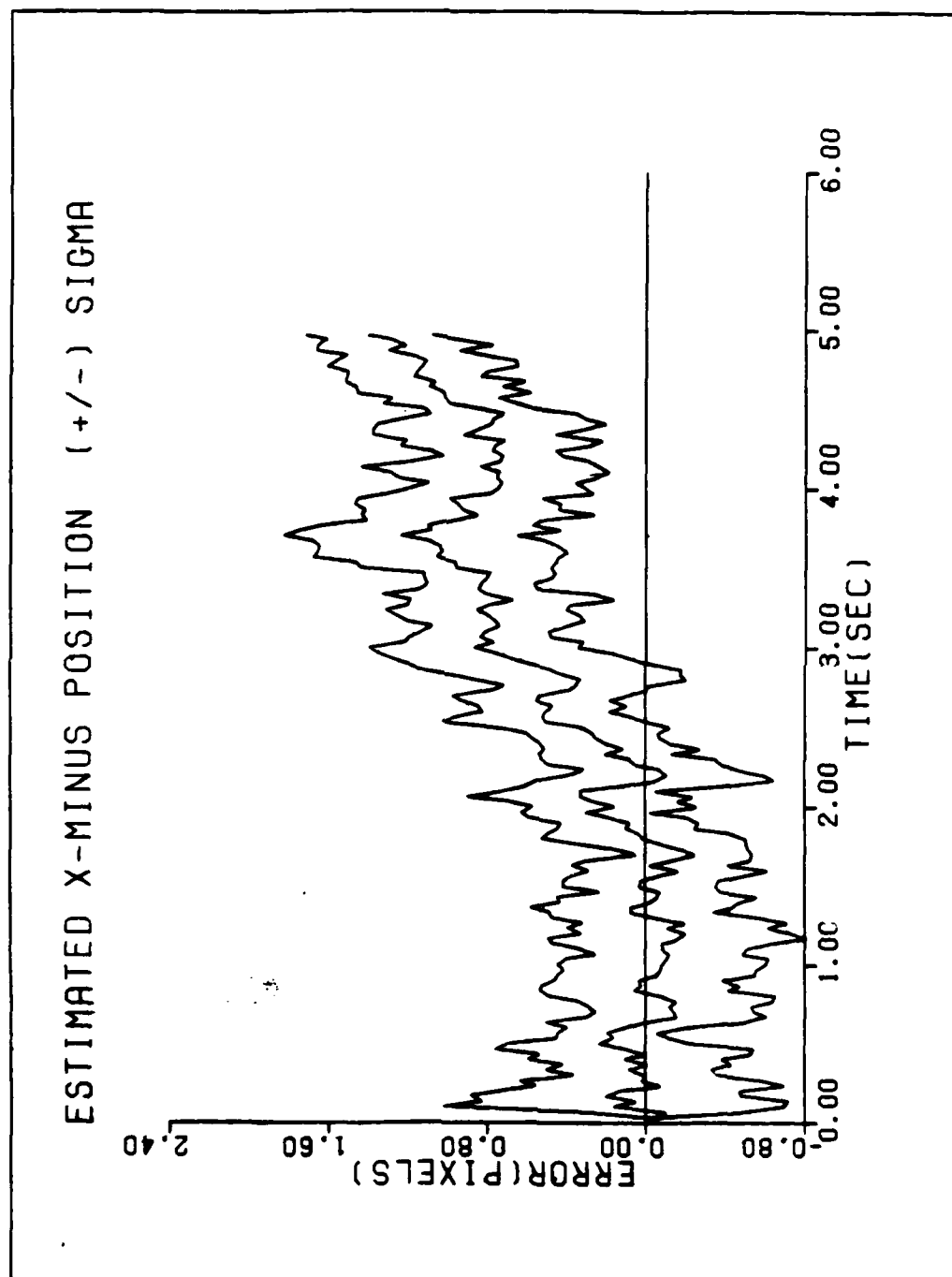


Figure A5 Performance plot for T2G20N

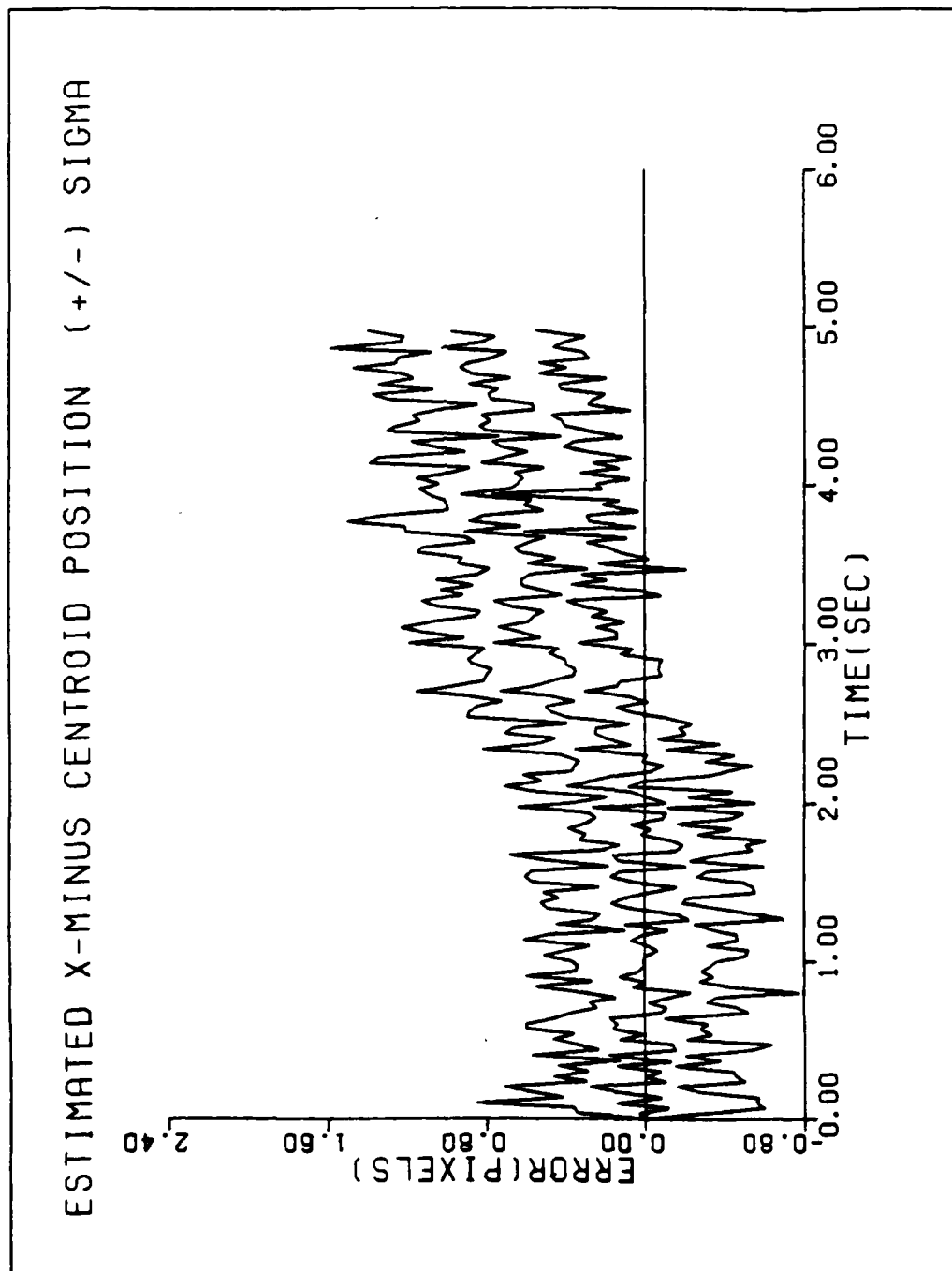


Figure A6 Performance plot for T2G20N

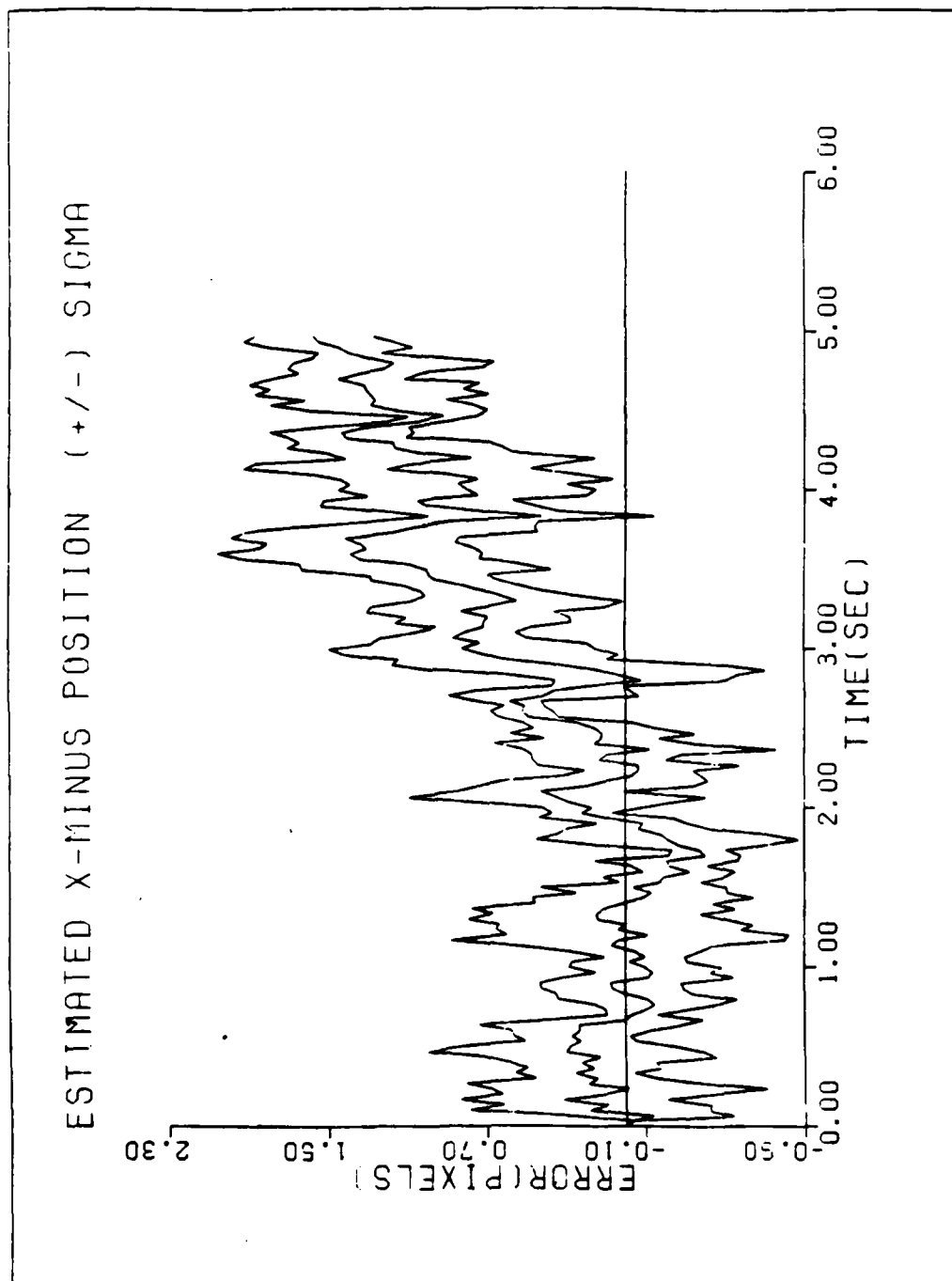


Figure A7 Performance plot for TINUM

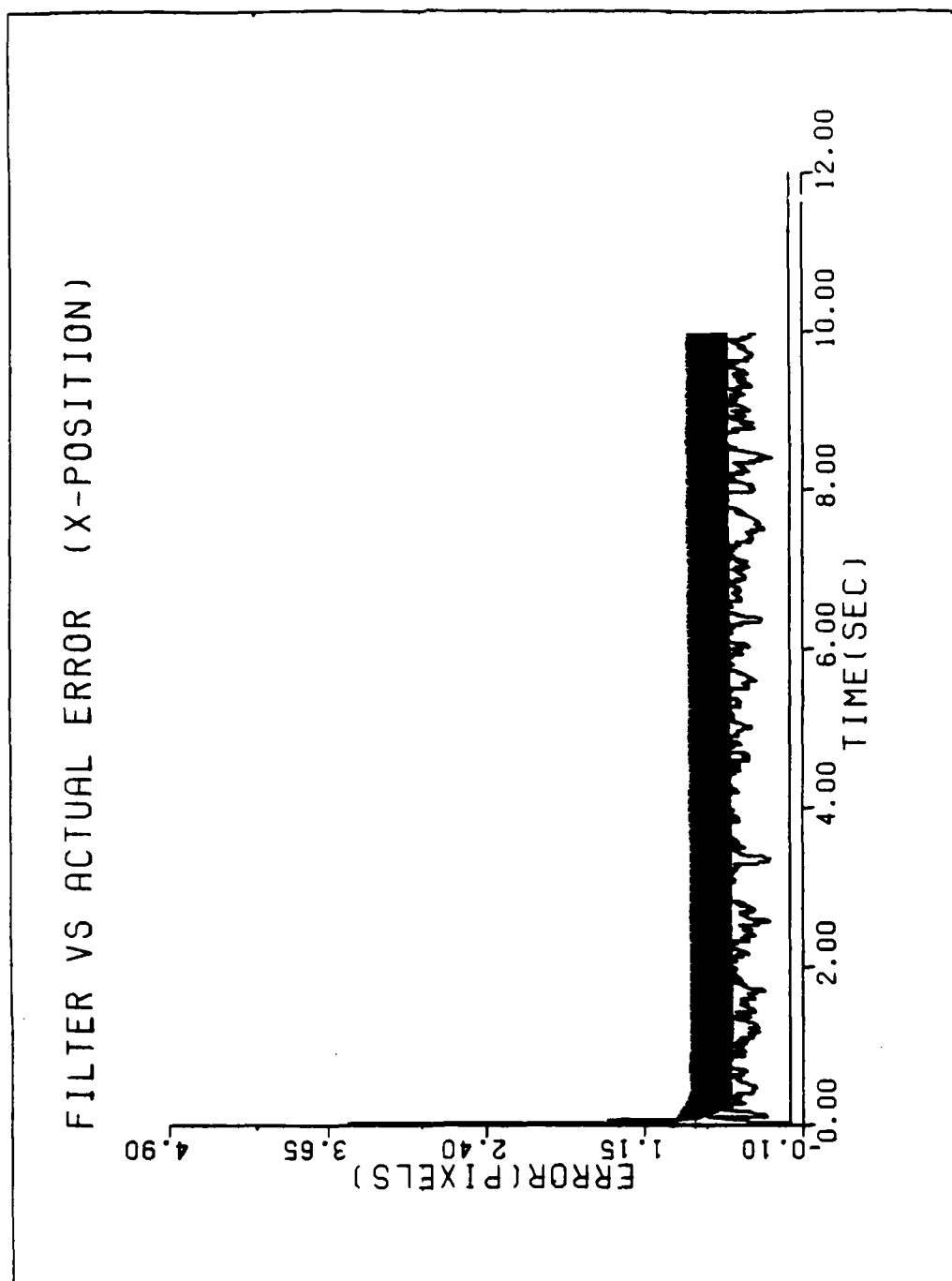


Figure A8 Performance plot for T1N

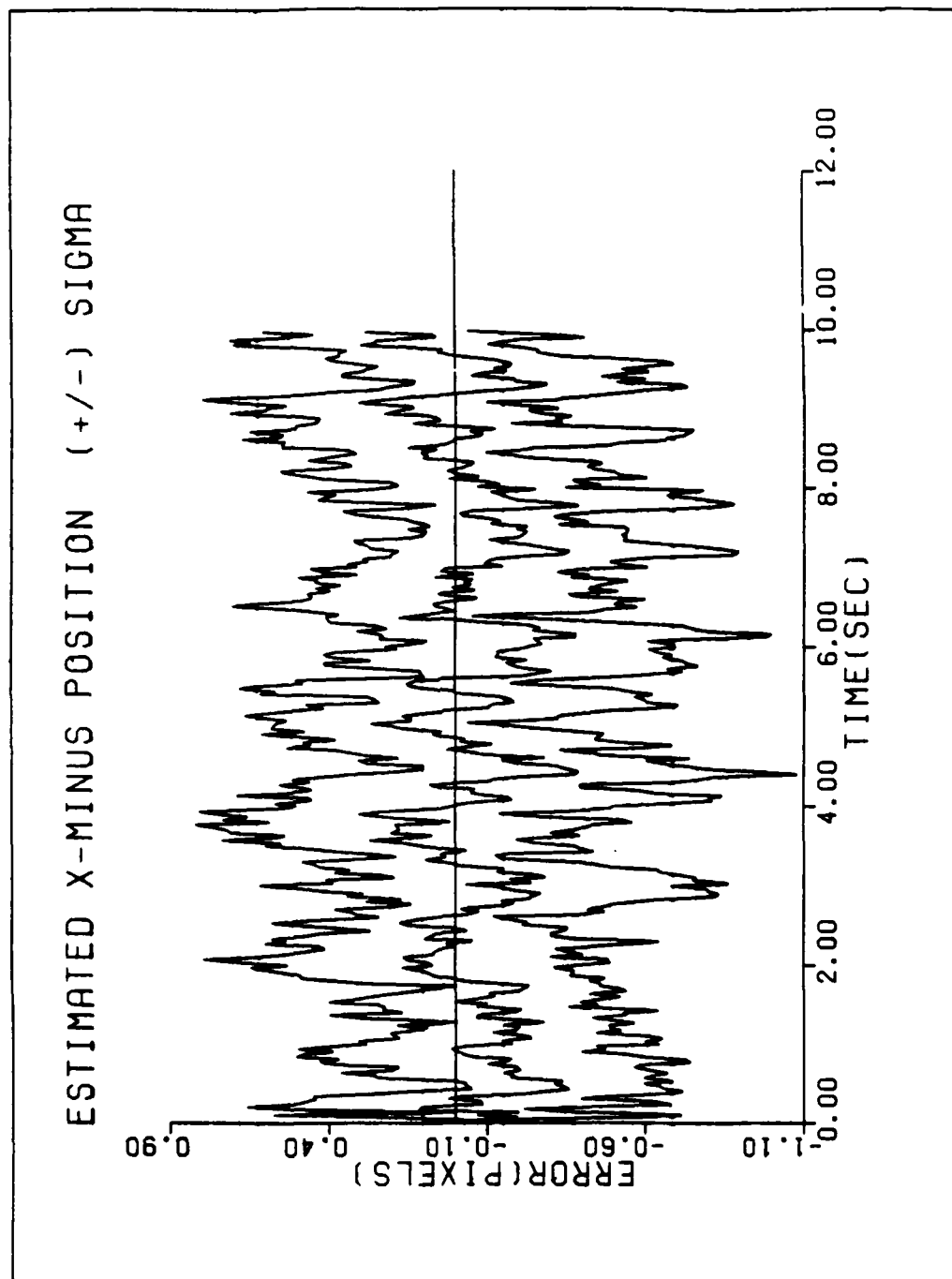


Figure A9 Performance plot for TIN

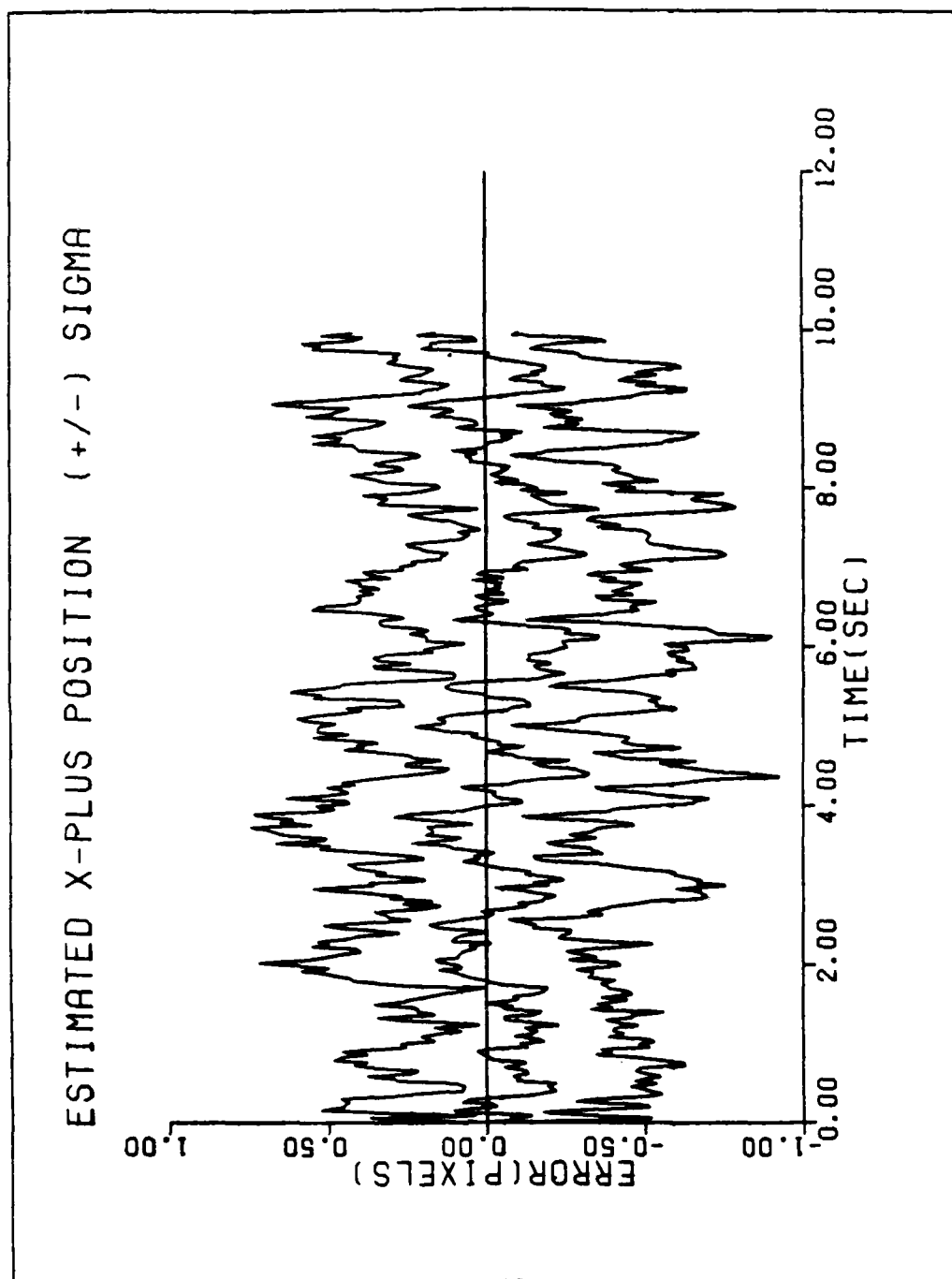


Figure A10 Performance plot for T1N

ESTIMATED X-MINUS CENTROID POSITION (+/-) SIGMA

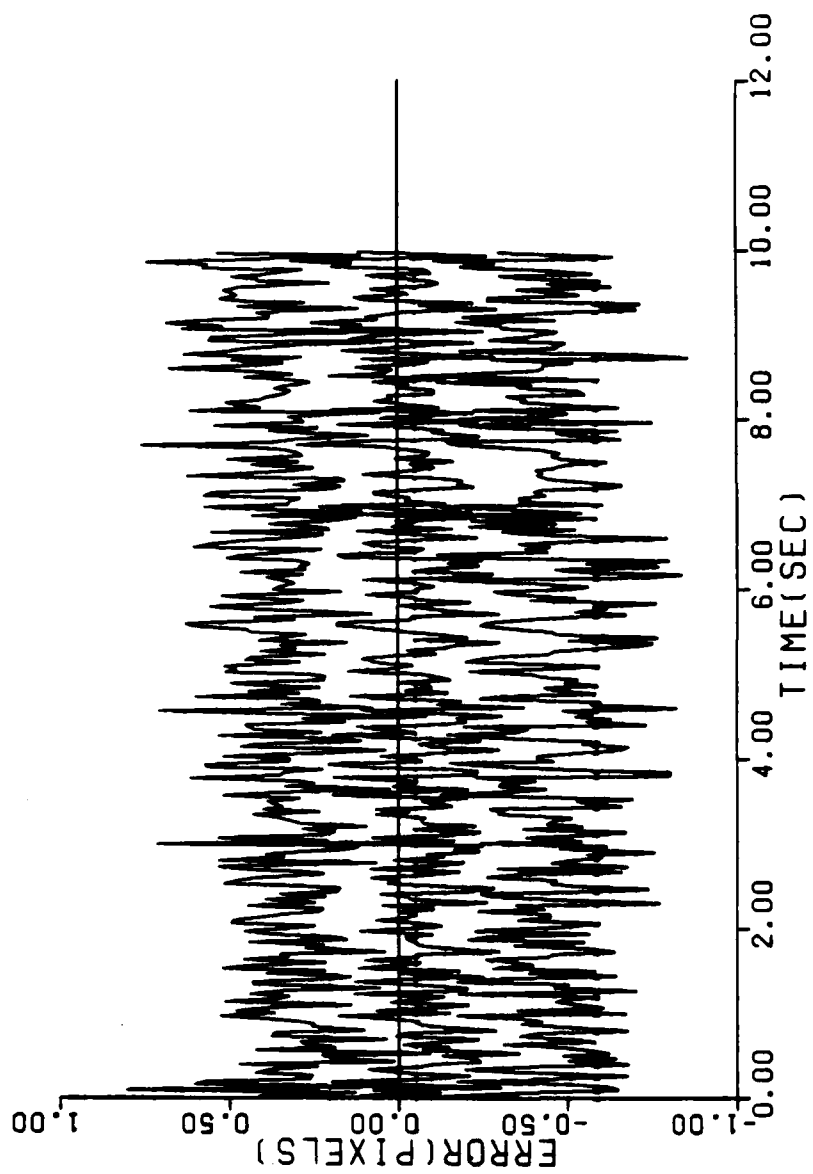


Figure A11 Performance plot for T1N

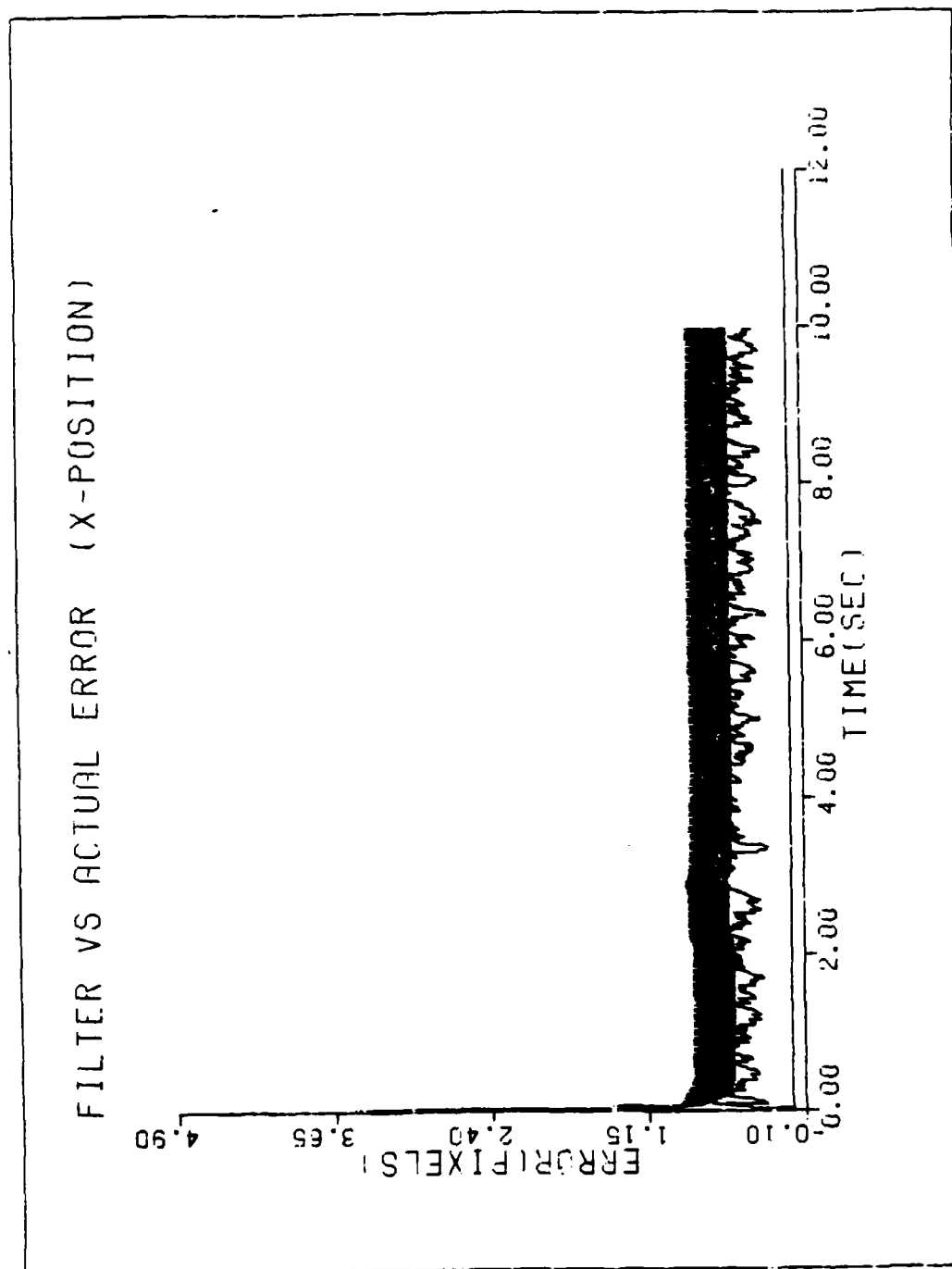


Figure A12 Performance plot for T2G10N

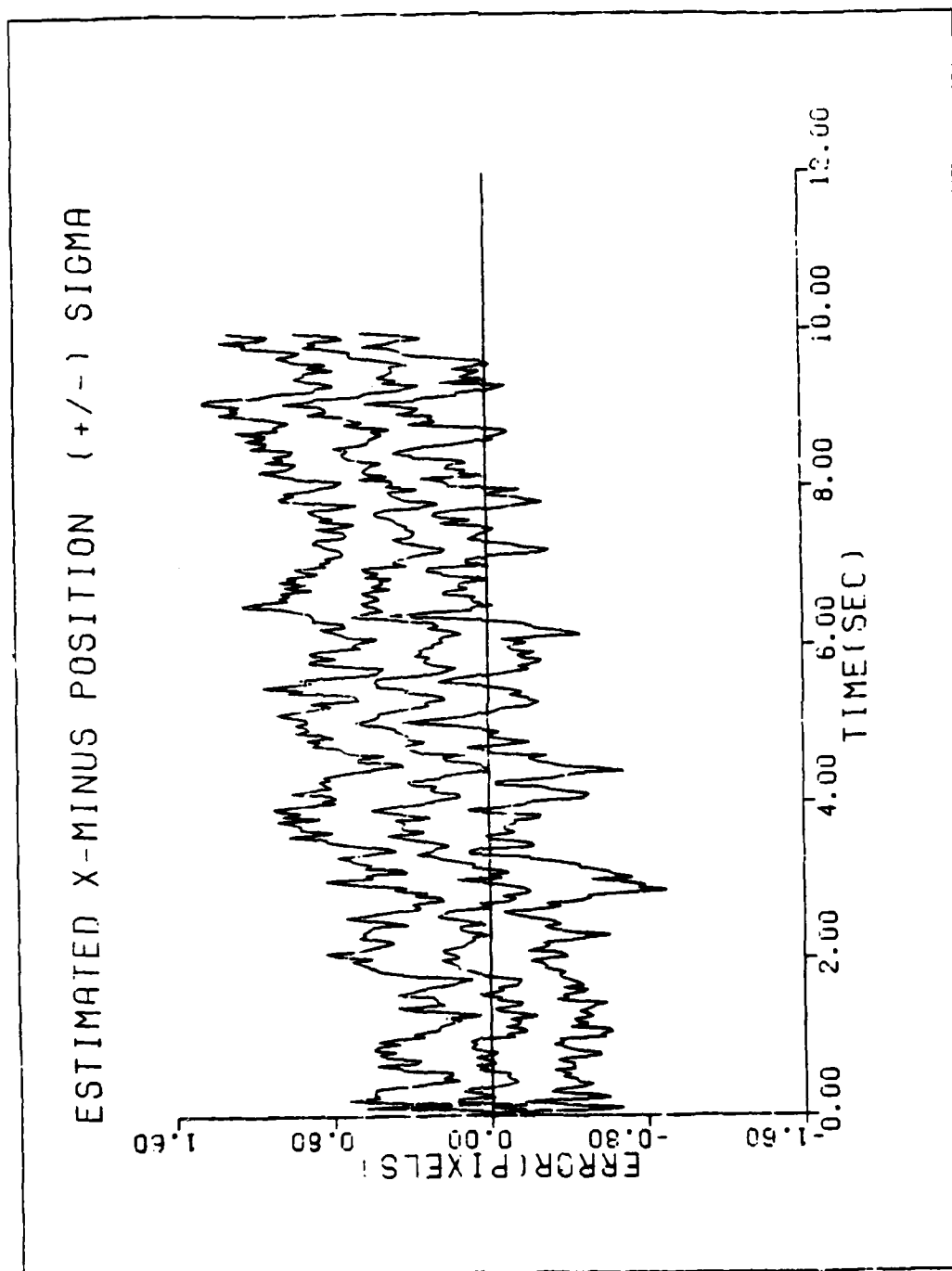


Figure A13 Performance plot for T2G10N

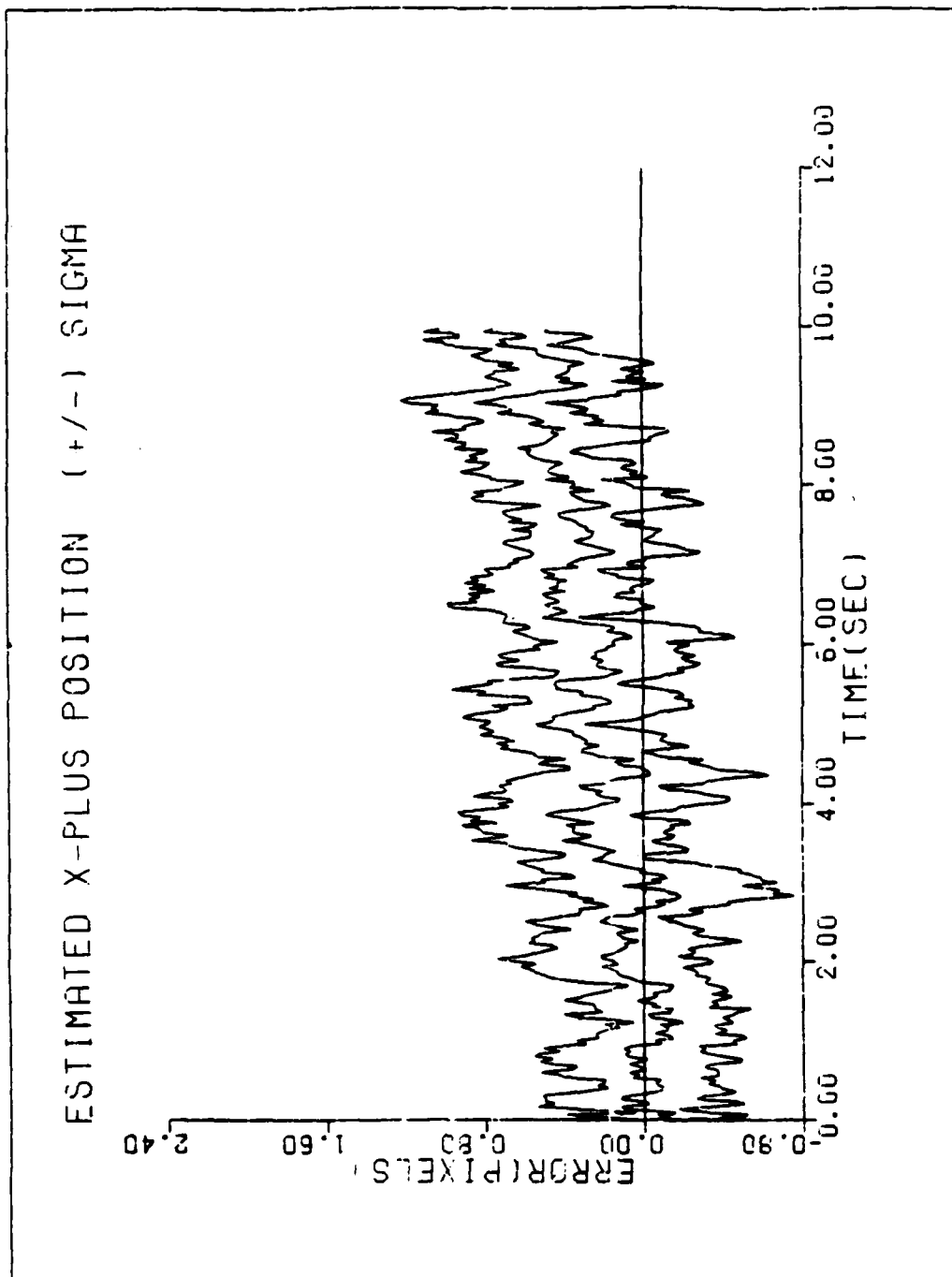


Figure A14 Performance plot for T2G10N

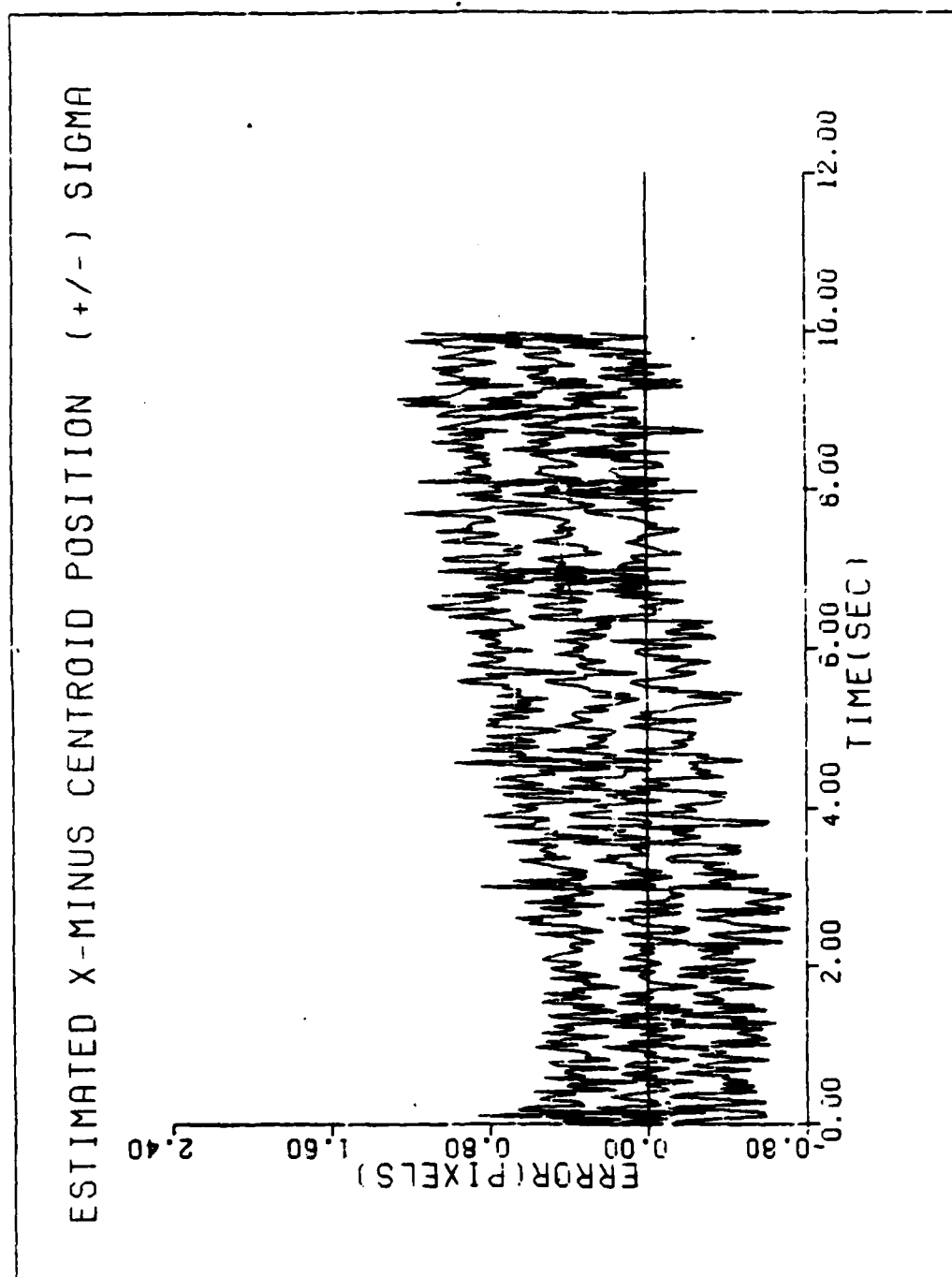


Figure A15 Performance plot for T2G10N

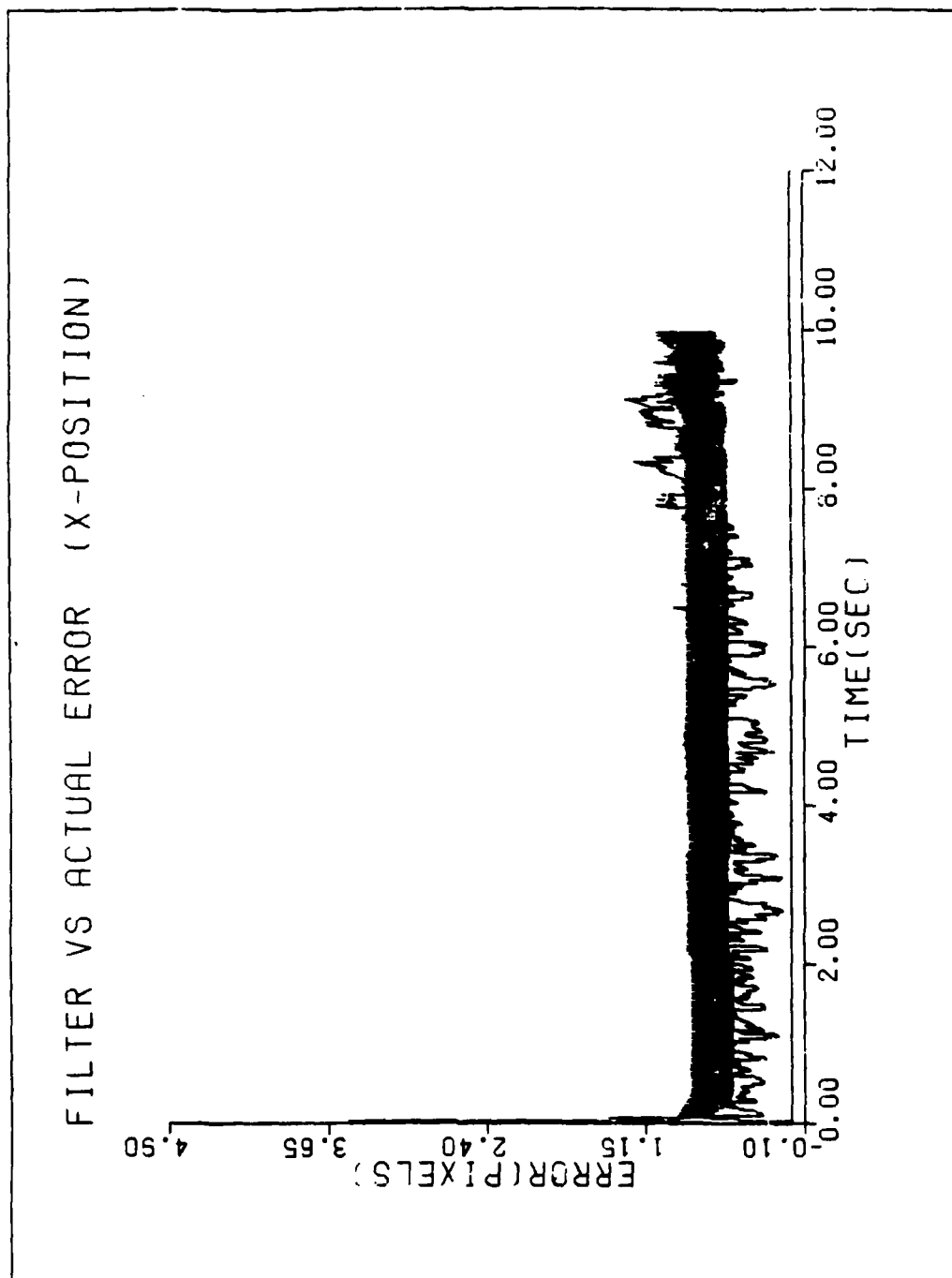


Figure A16 Performance plot for T2G20N

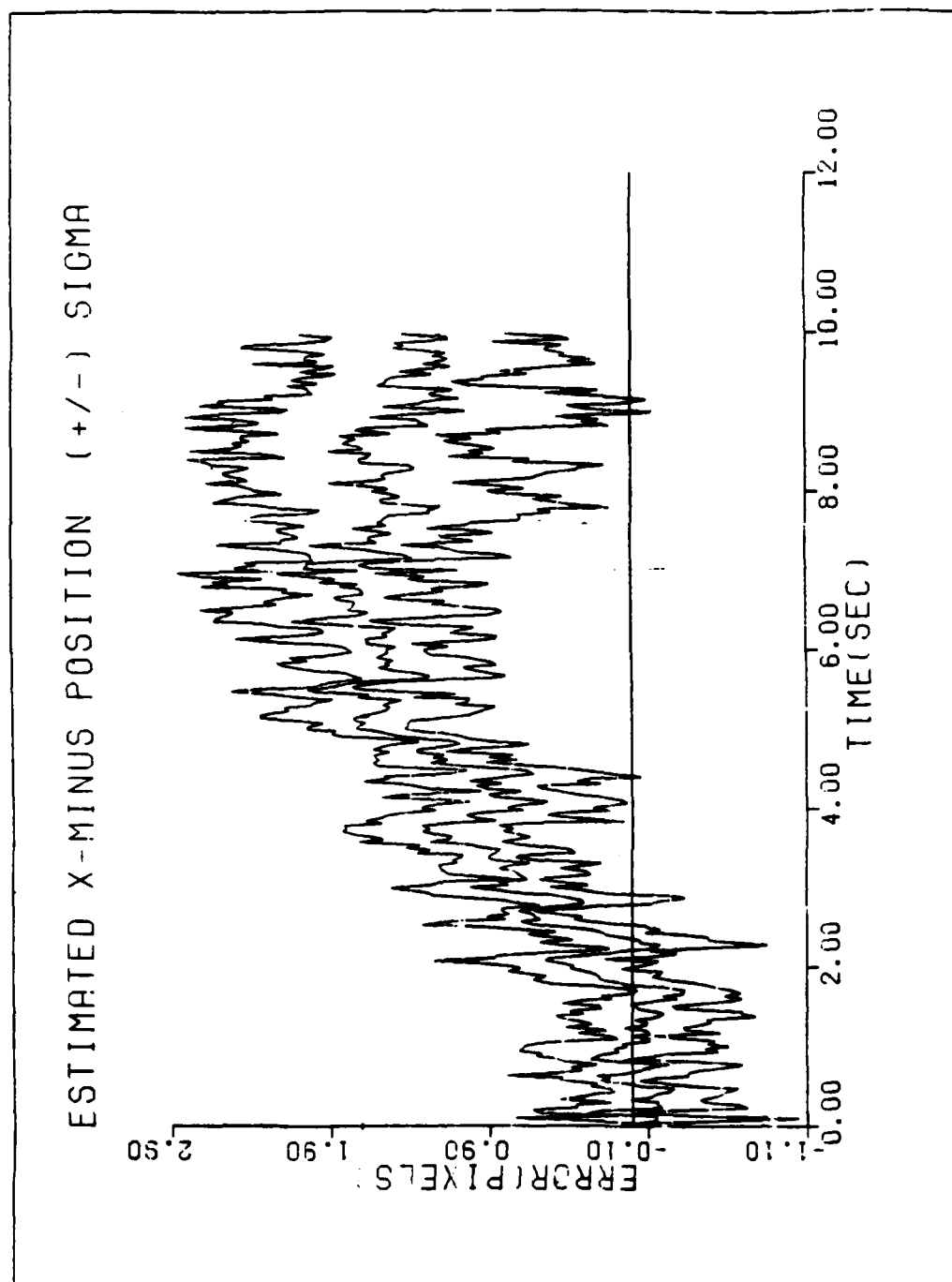


Figure A17 Performance plot for T2G20N

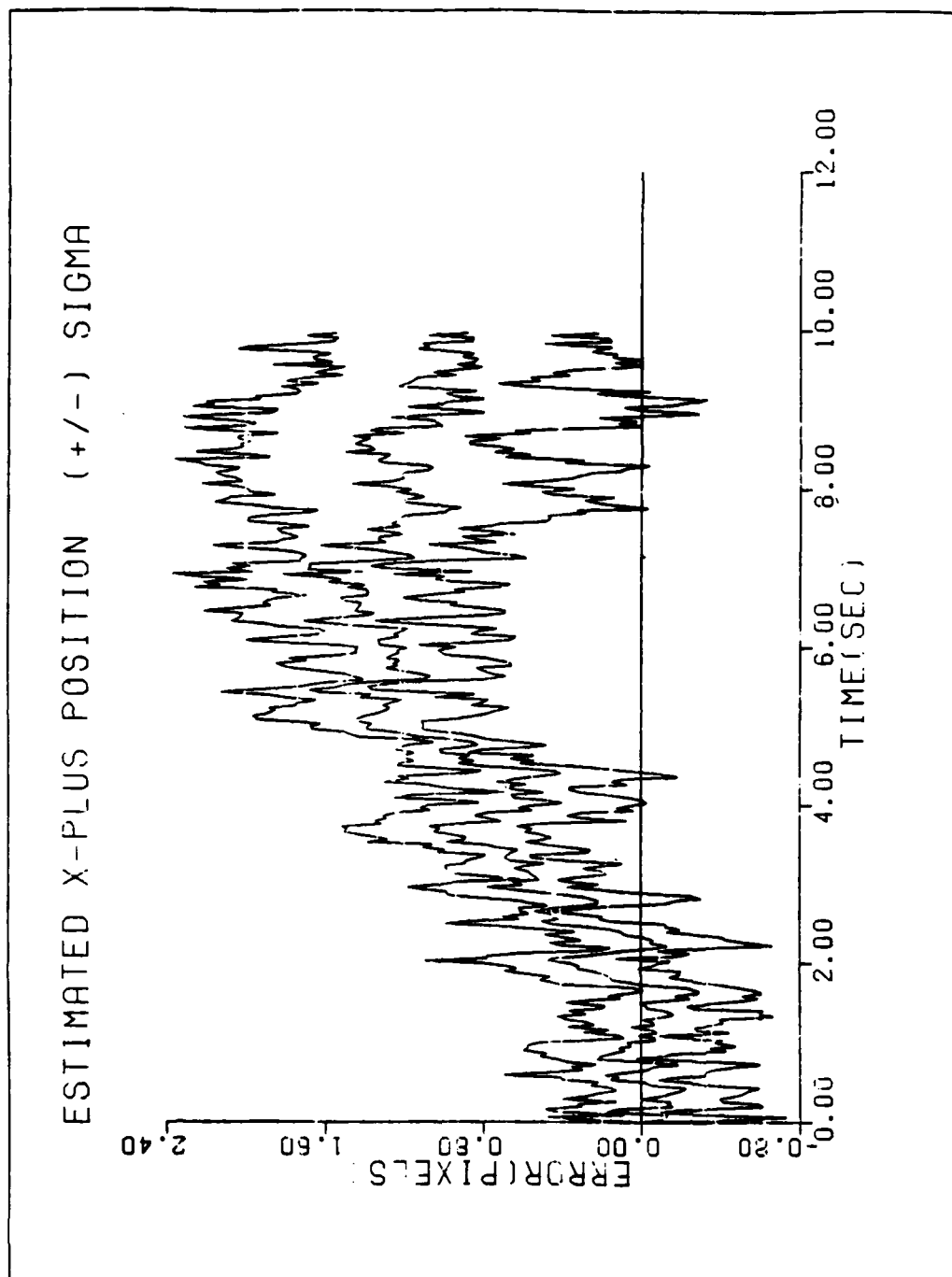


Figure A18 Performance plot for T2G20N

ESTIMATED X-MINUS CENTROID POSITION (+/-) SIGMA

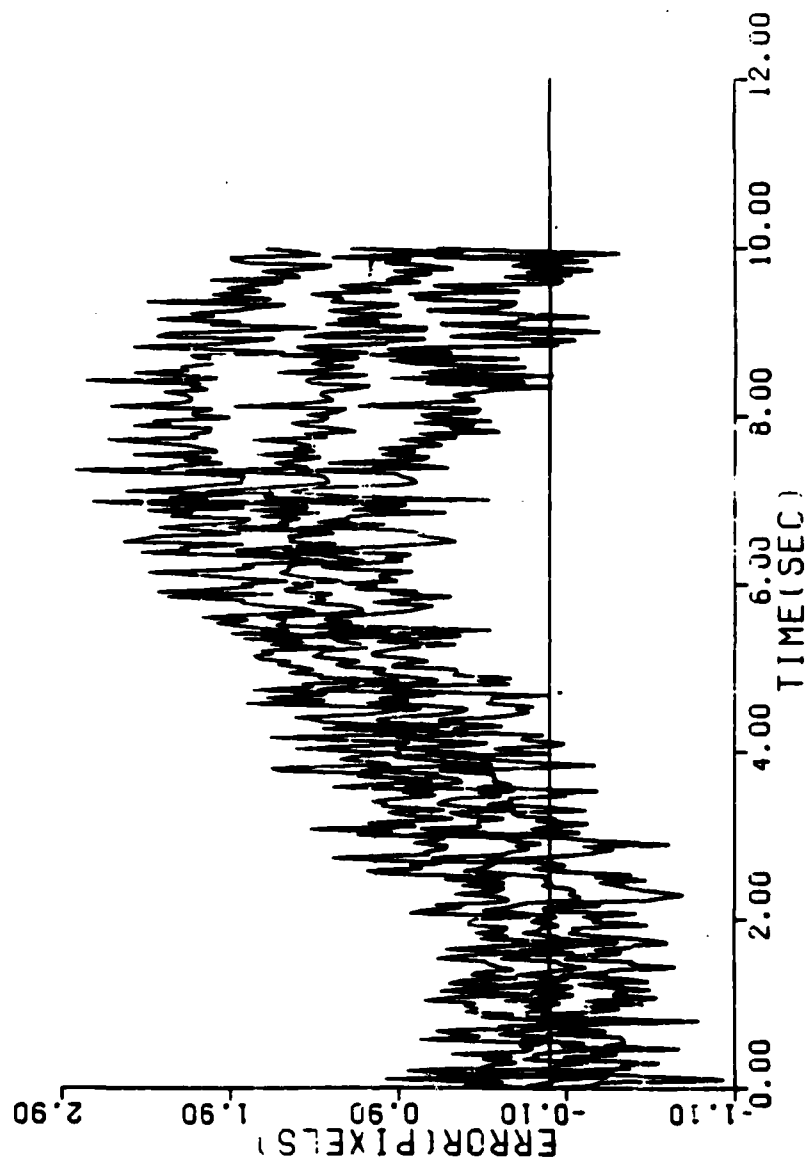


Figure A19 Performance plot for T2G20N

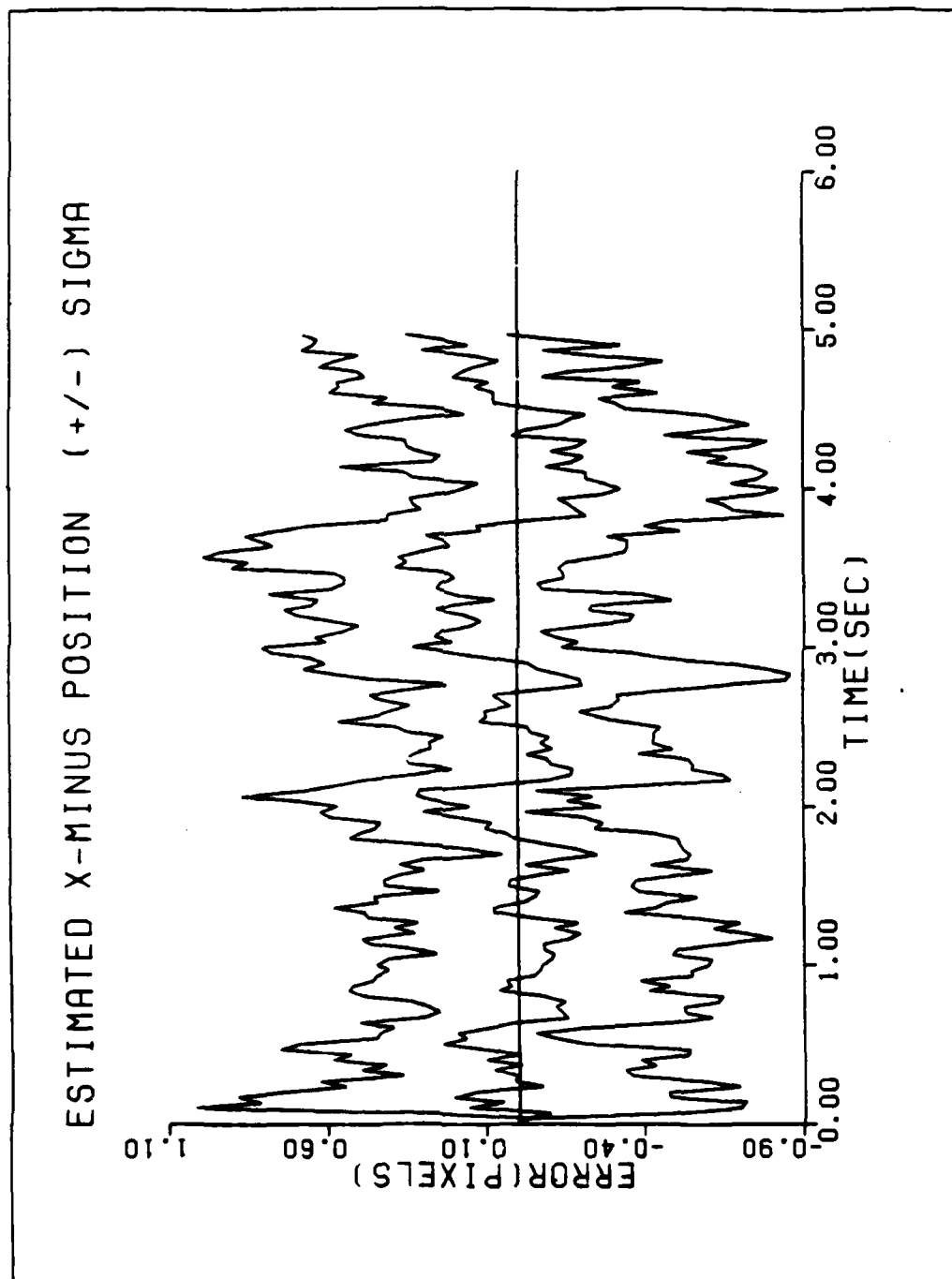


Figure A20 Performance plot for T3G10N

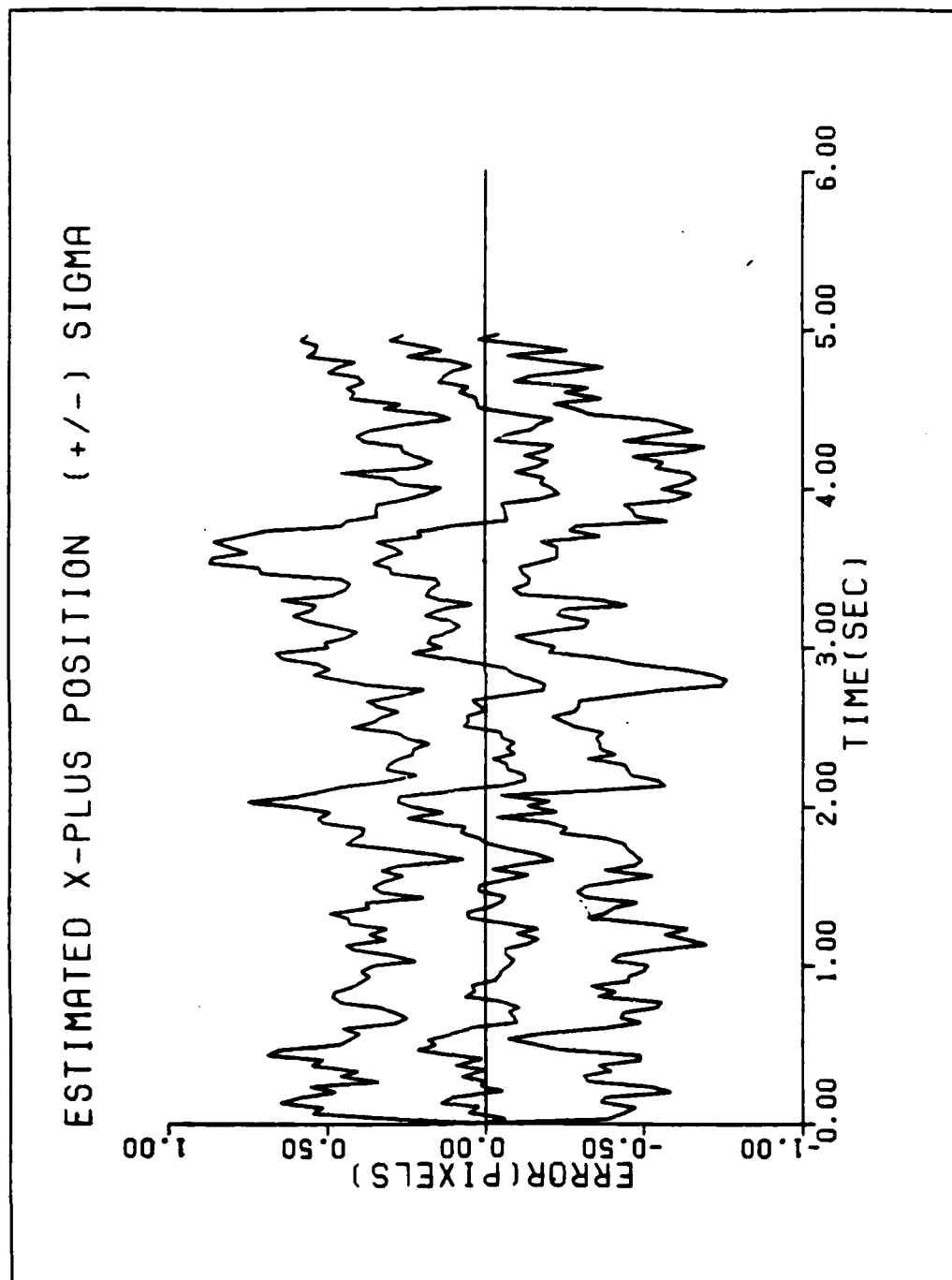


Figure A21 Performance plot for T3G10N

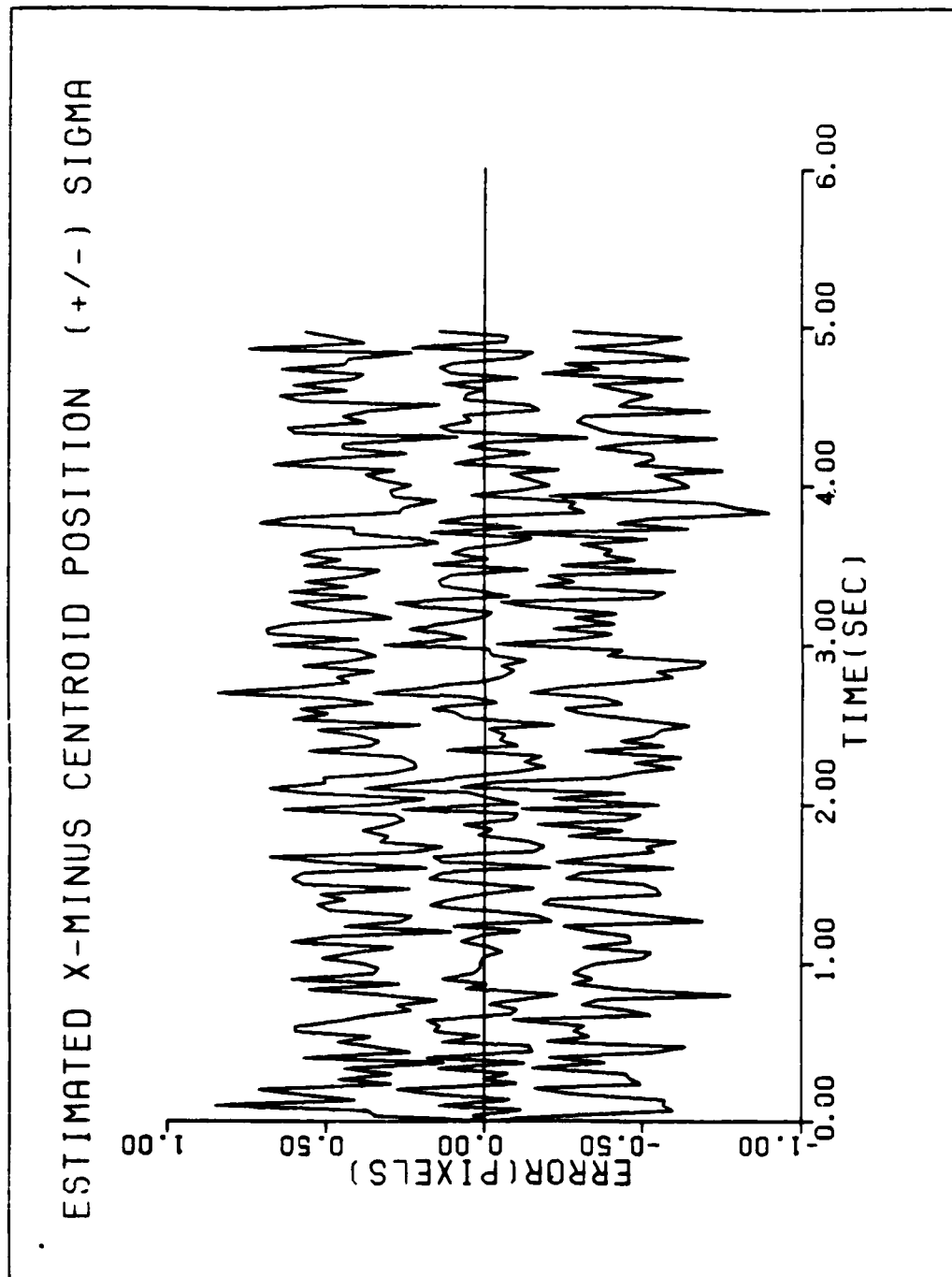


Figure A22 Performance plot for T3G10N

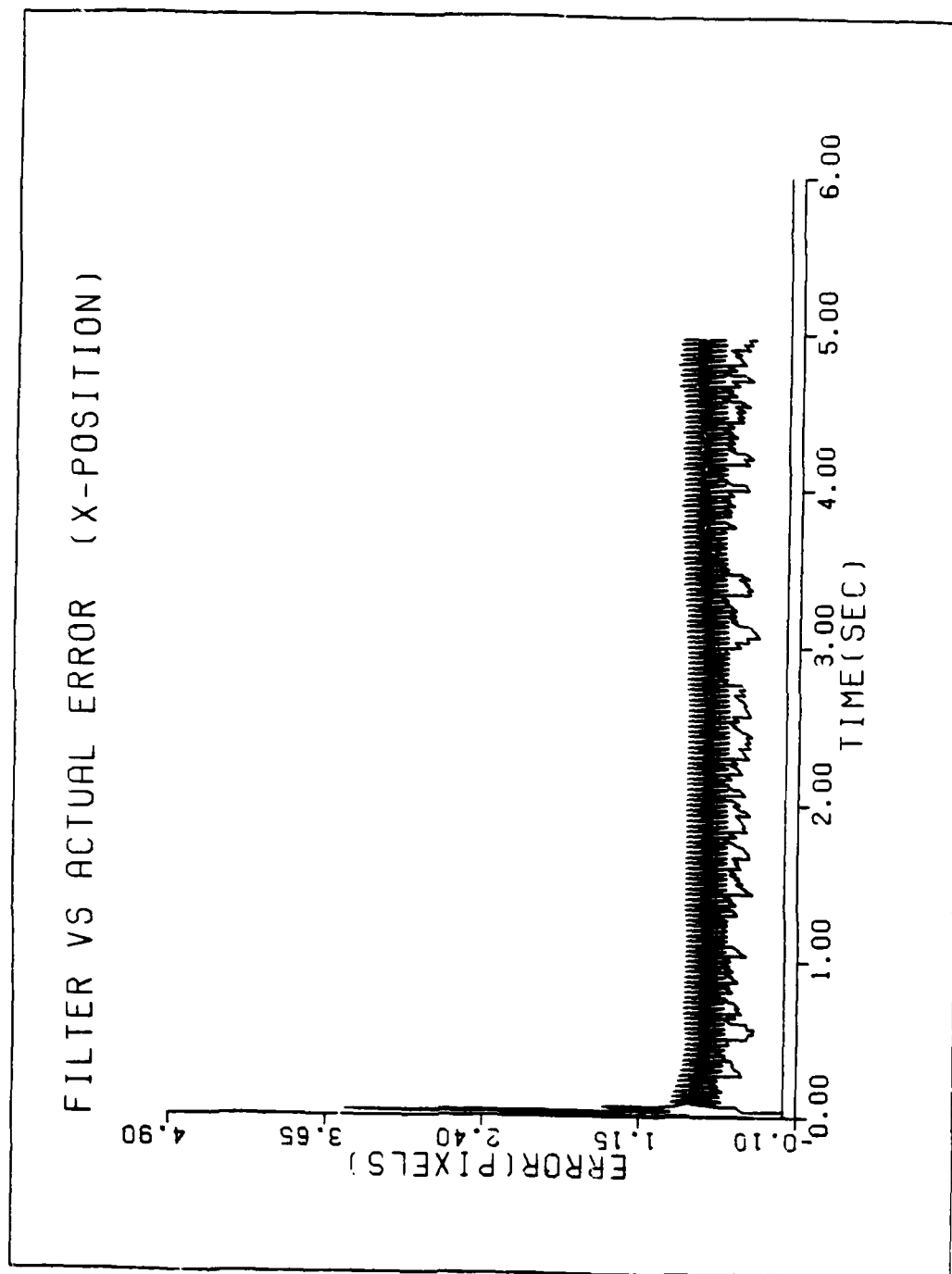


Figure A23 Performance plot for T2G20IF

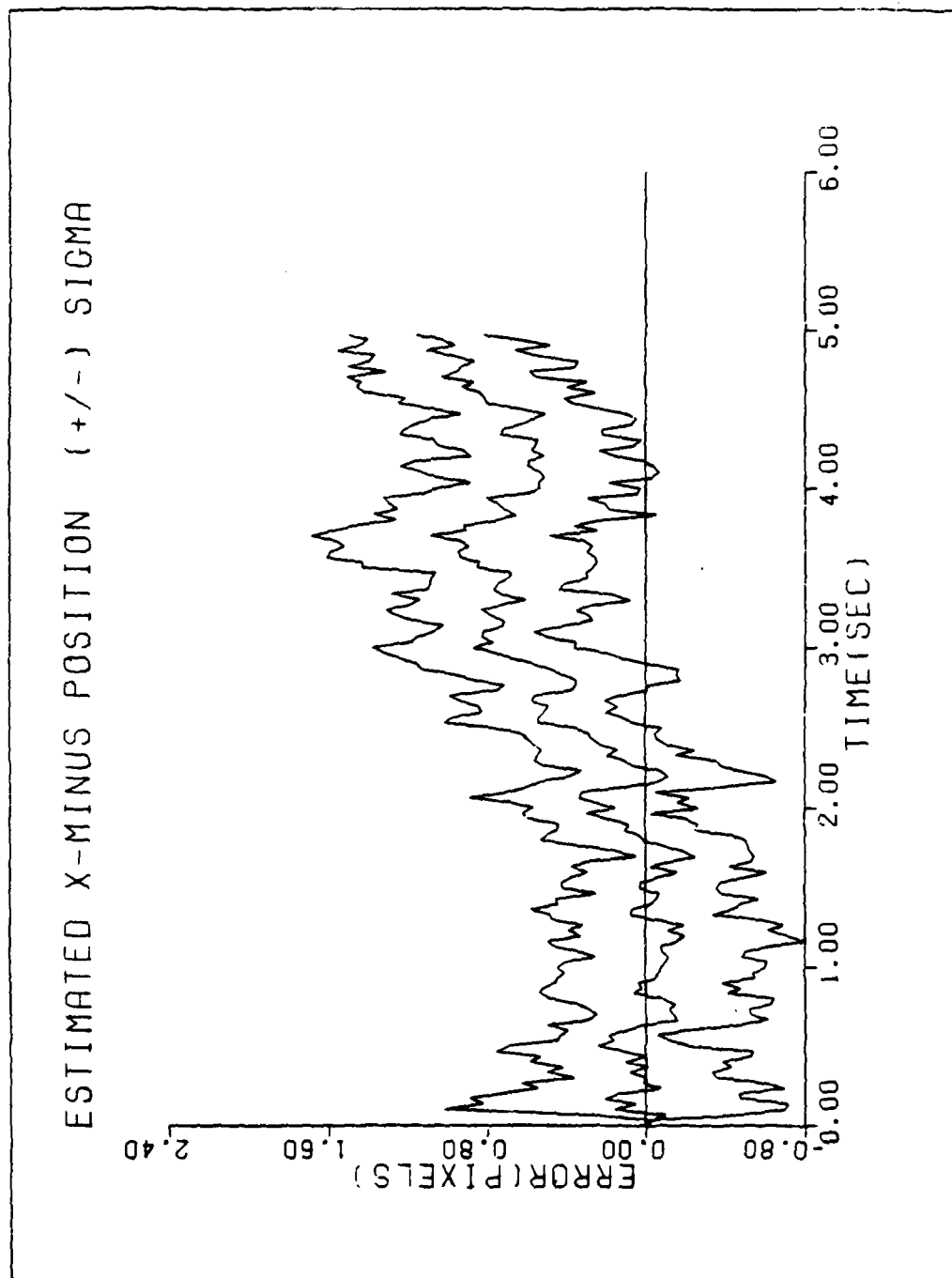


Figure A24 Performance plot for T2G20IF

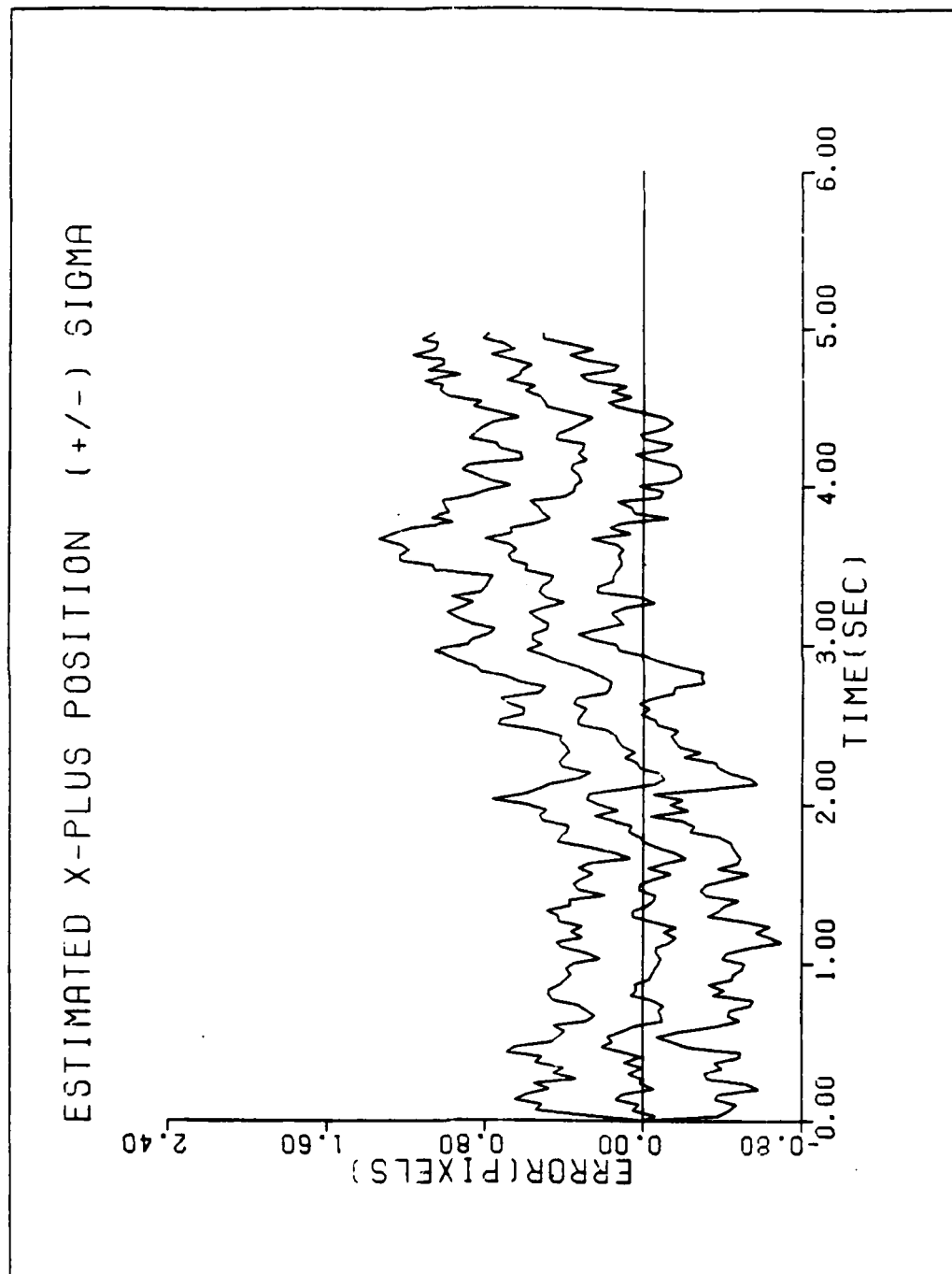


Figure A25 Performance plot for T2G20IF

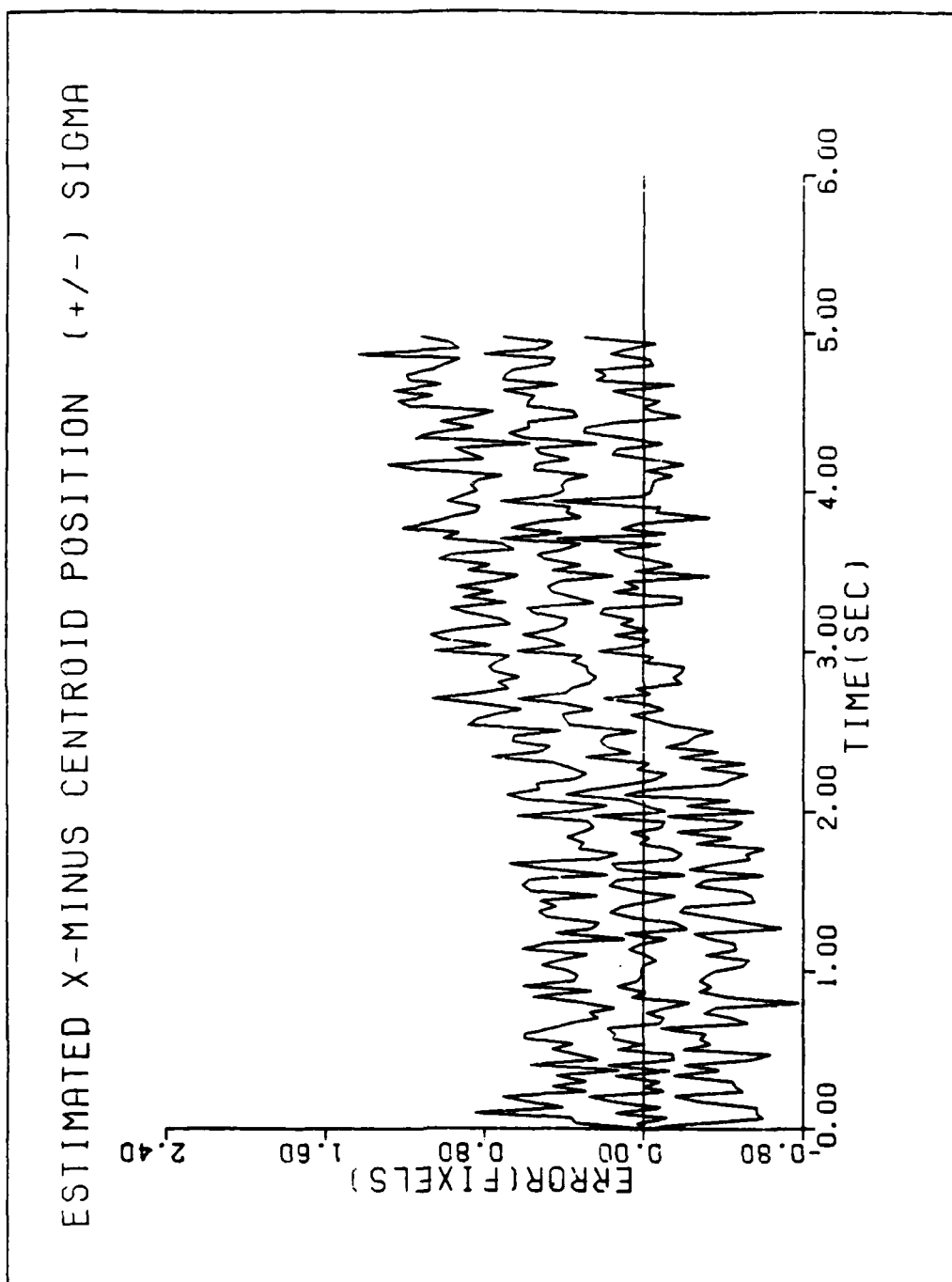


Figure A26 Performance plot for T2G20IF

FILTER VS ACTUAL ERROR (Y-POSITION)

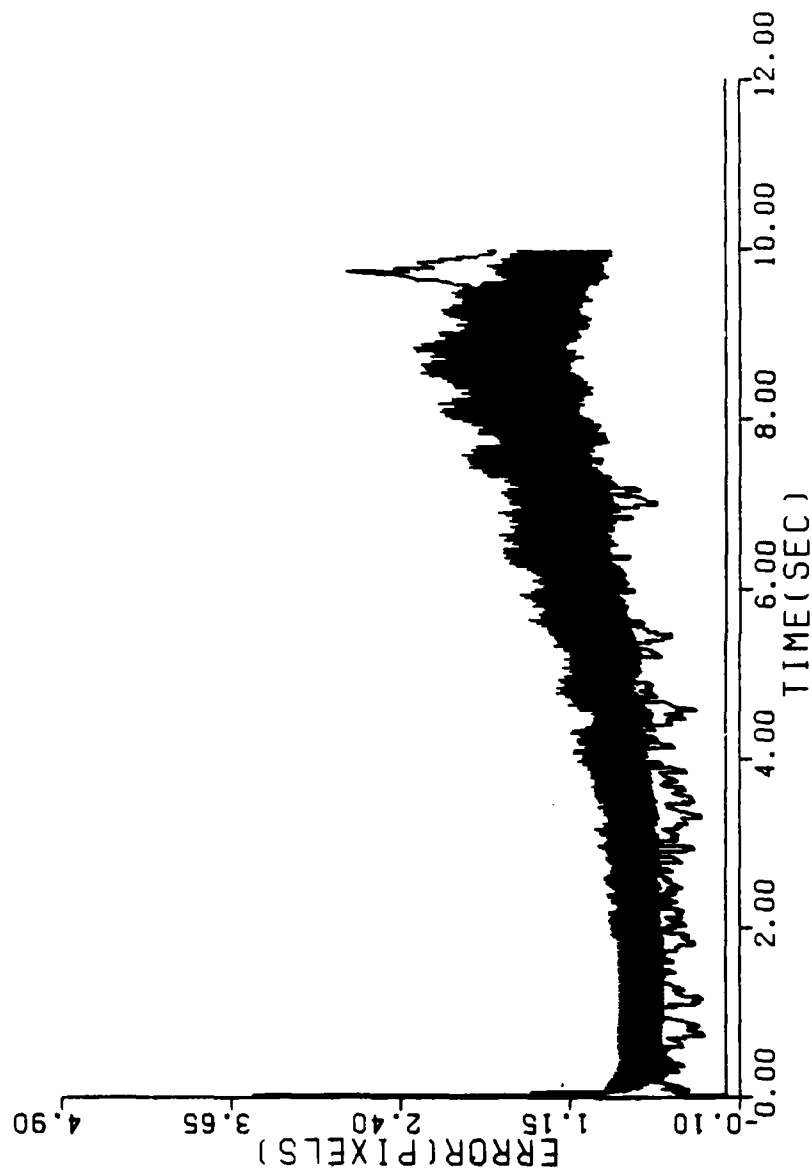


Figure A27 Performance plot for TiN

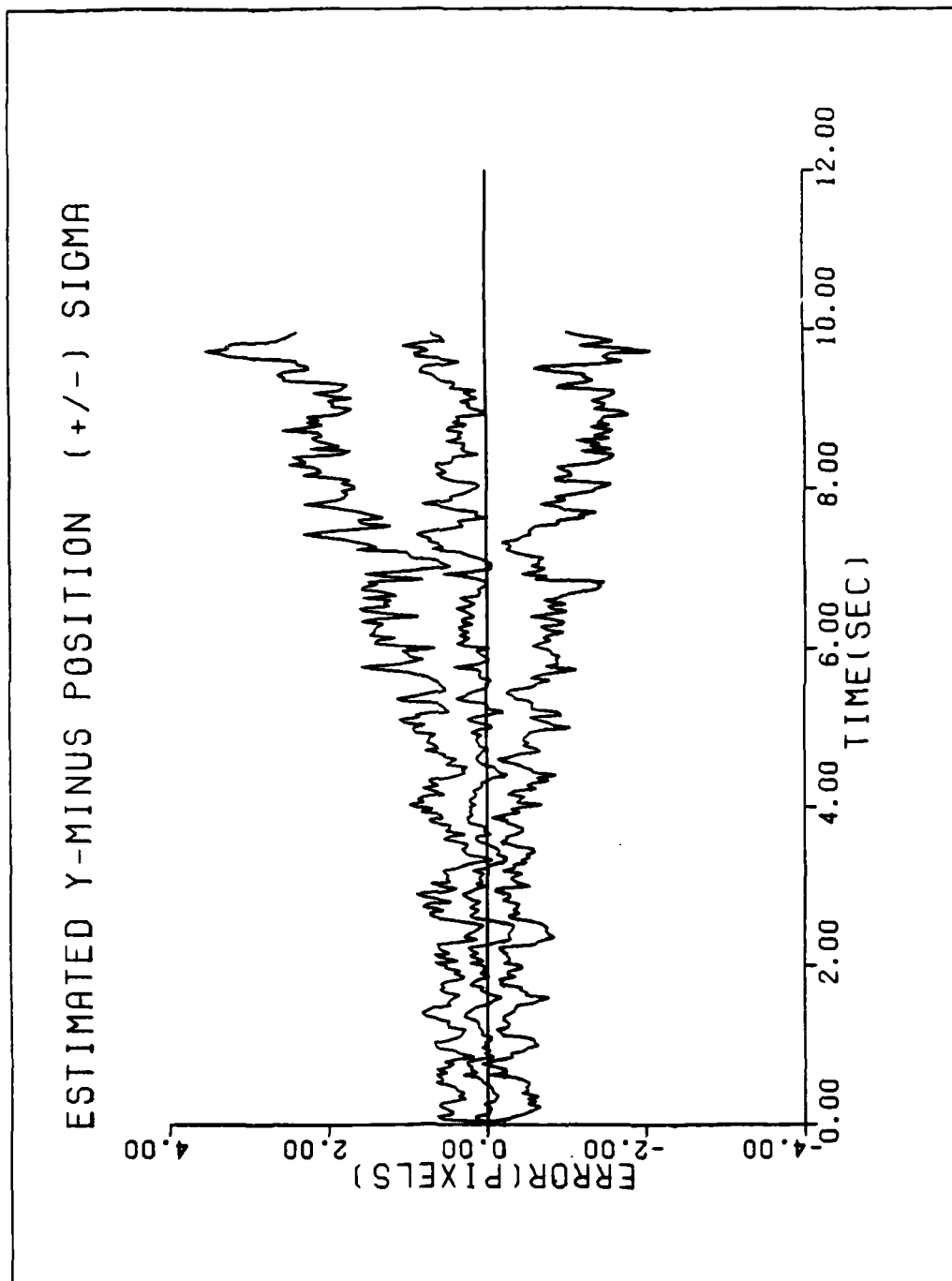


Figure A28 Performance plot for T1N

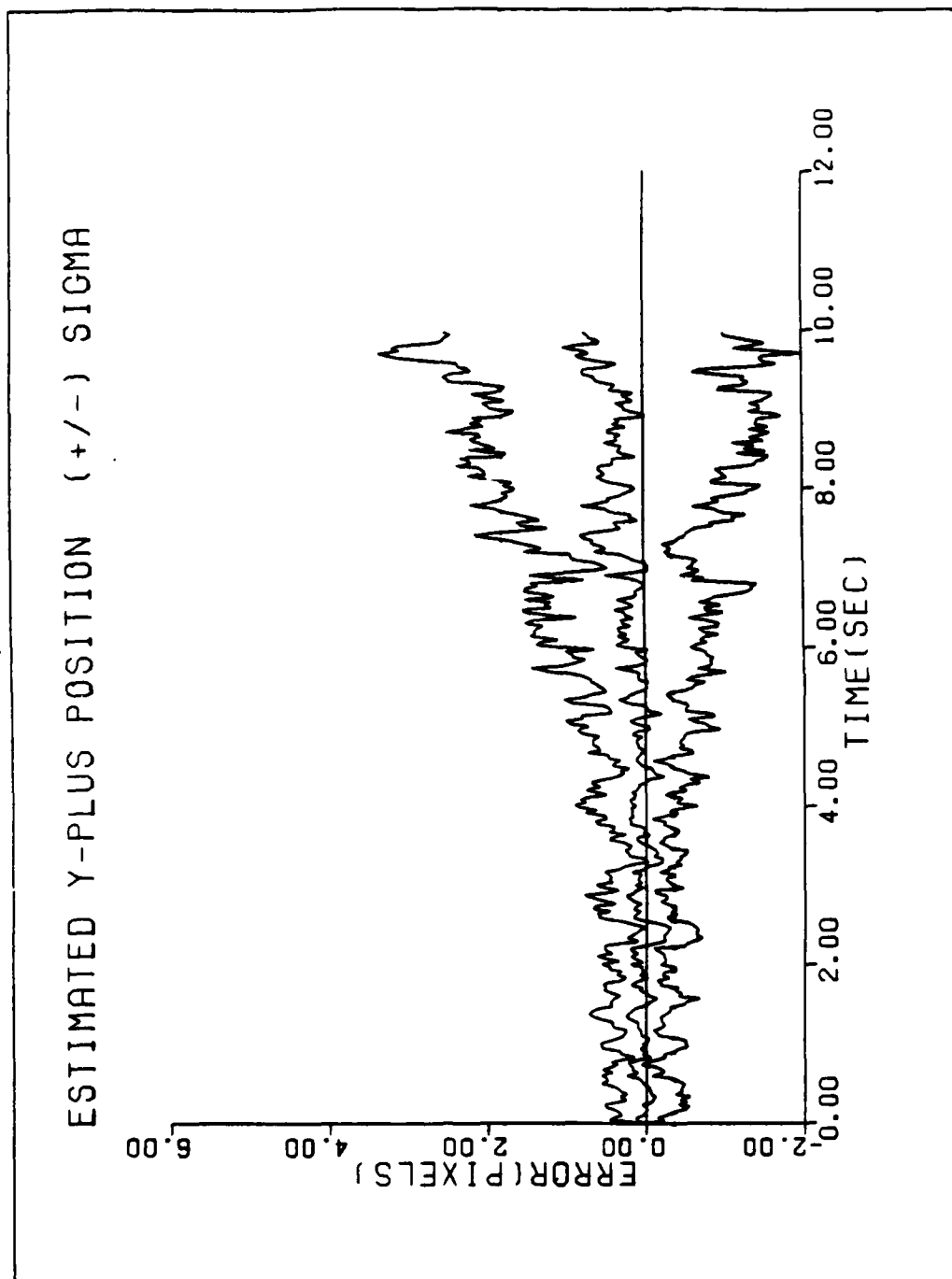


Figure A29 Performance plot for T1N

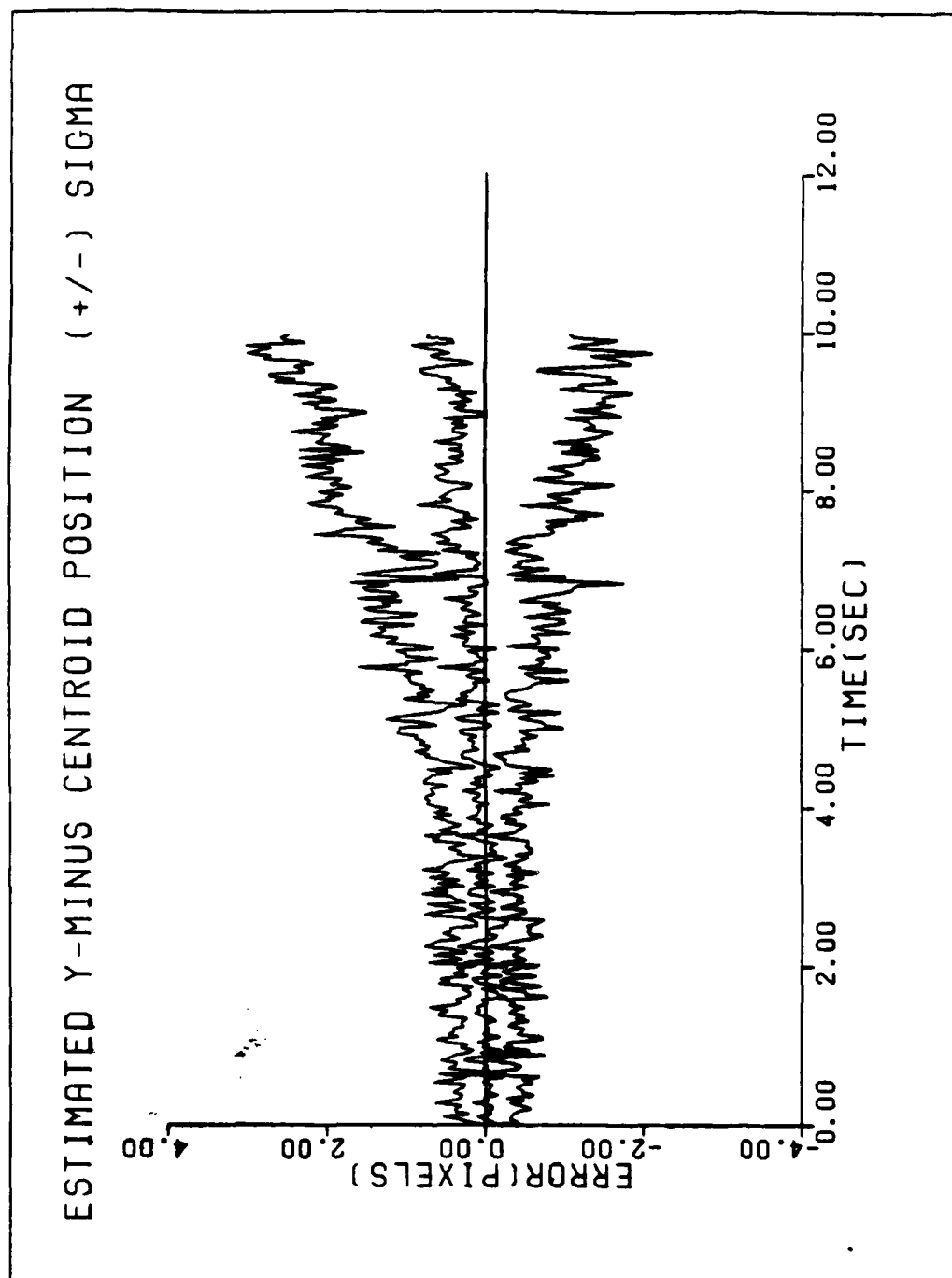


Figure A30 Performance plot for T1N

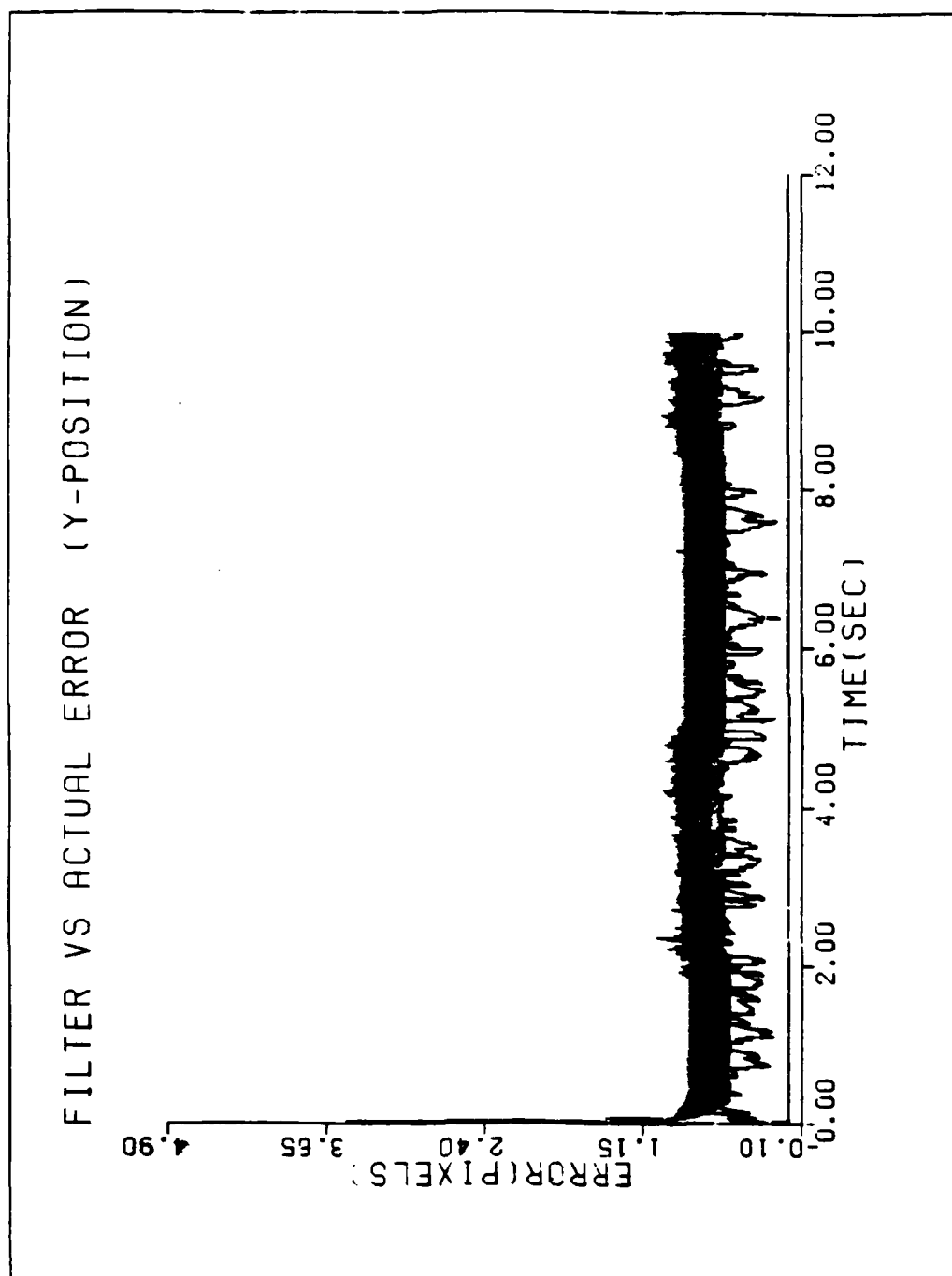


Figure A31 Performance plot for T2G20N

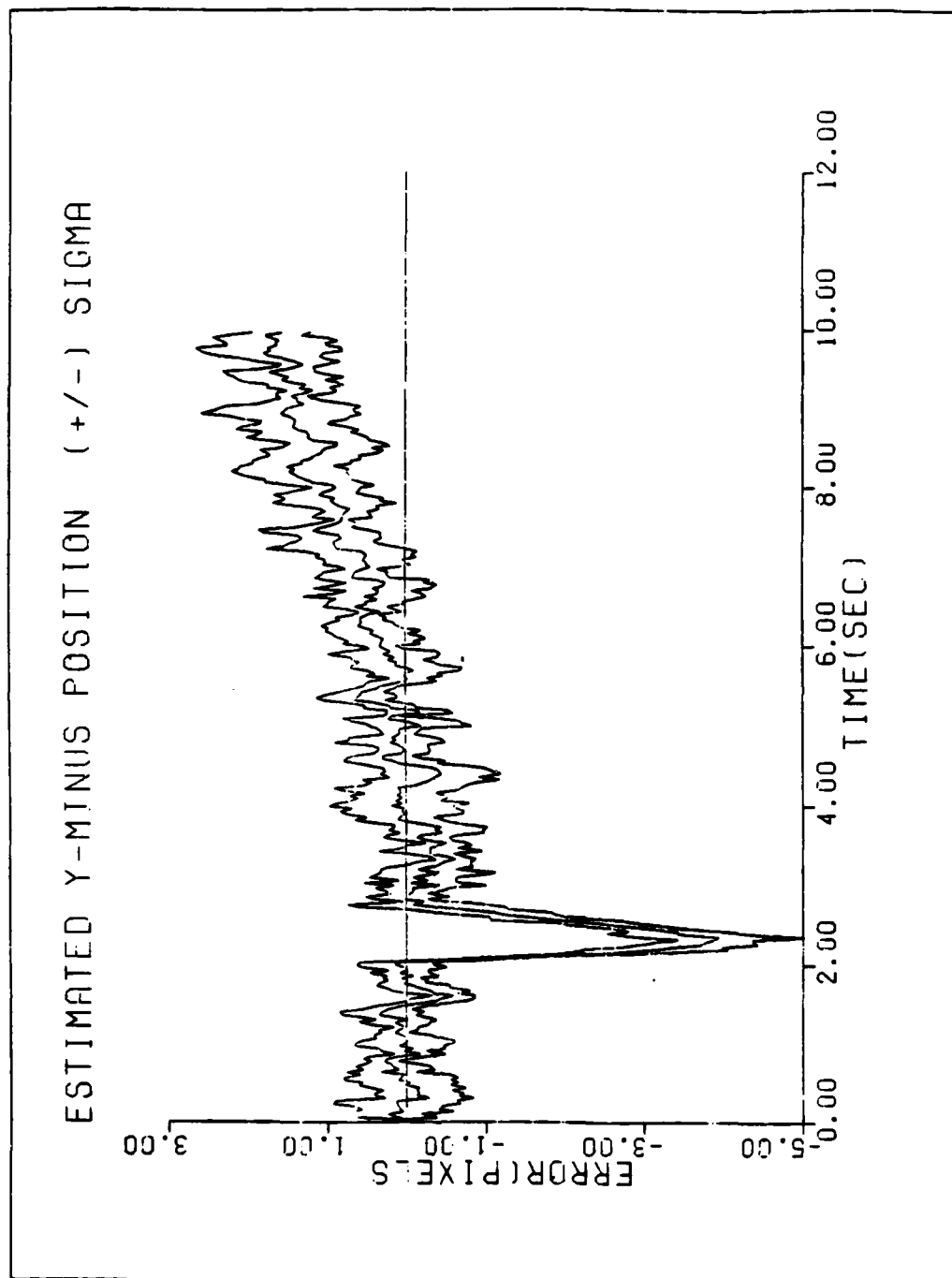


Figure A32 Performance plot for T2G20N

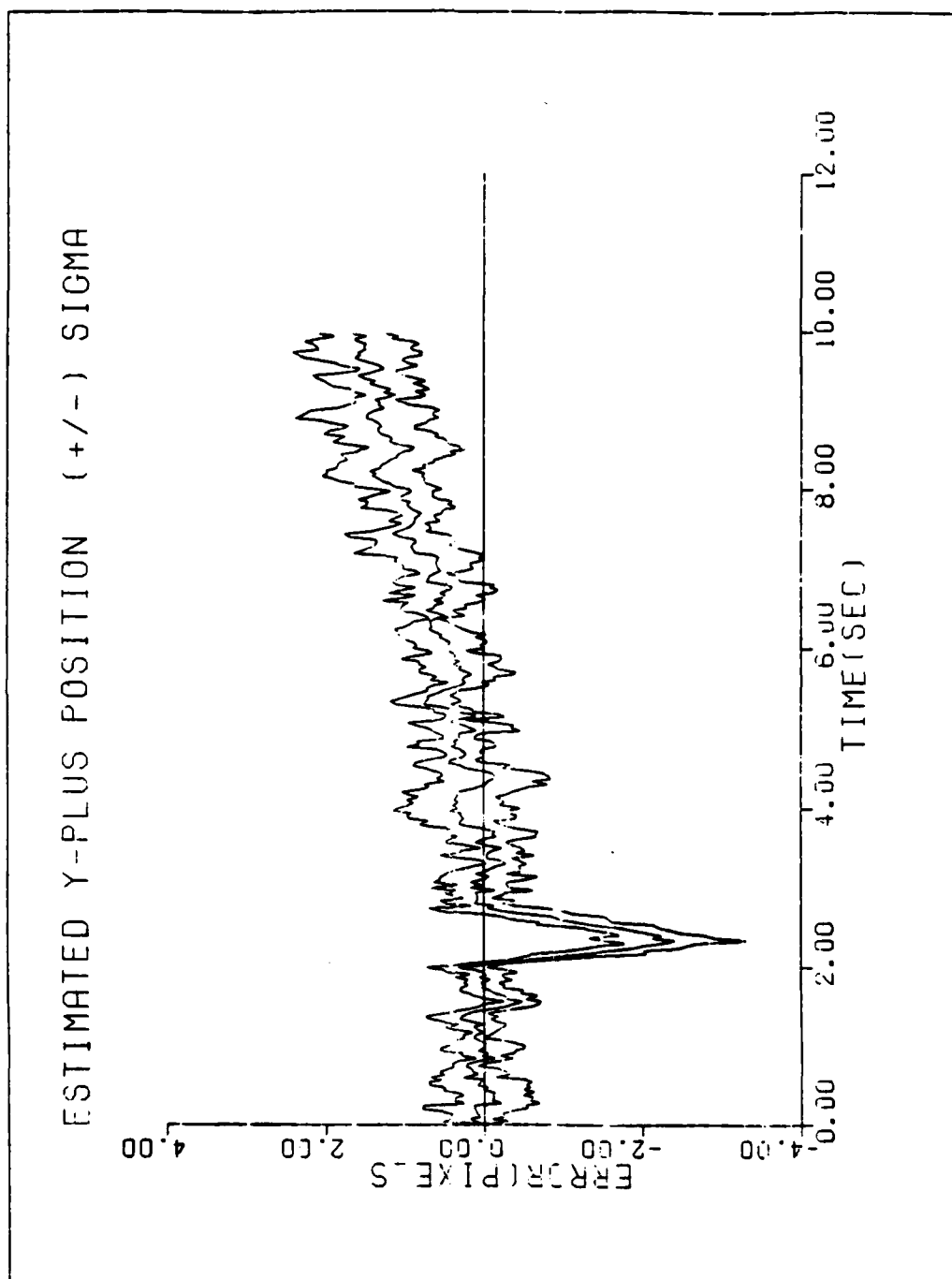


Figure A33 Performance plot for T2G20N

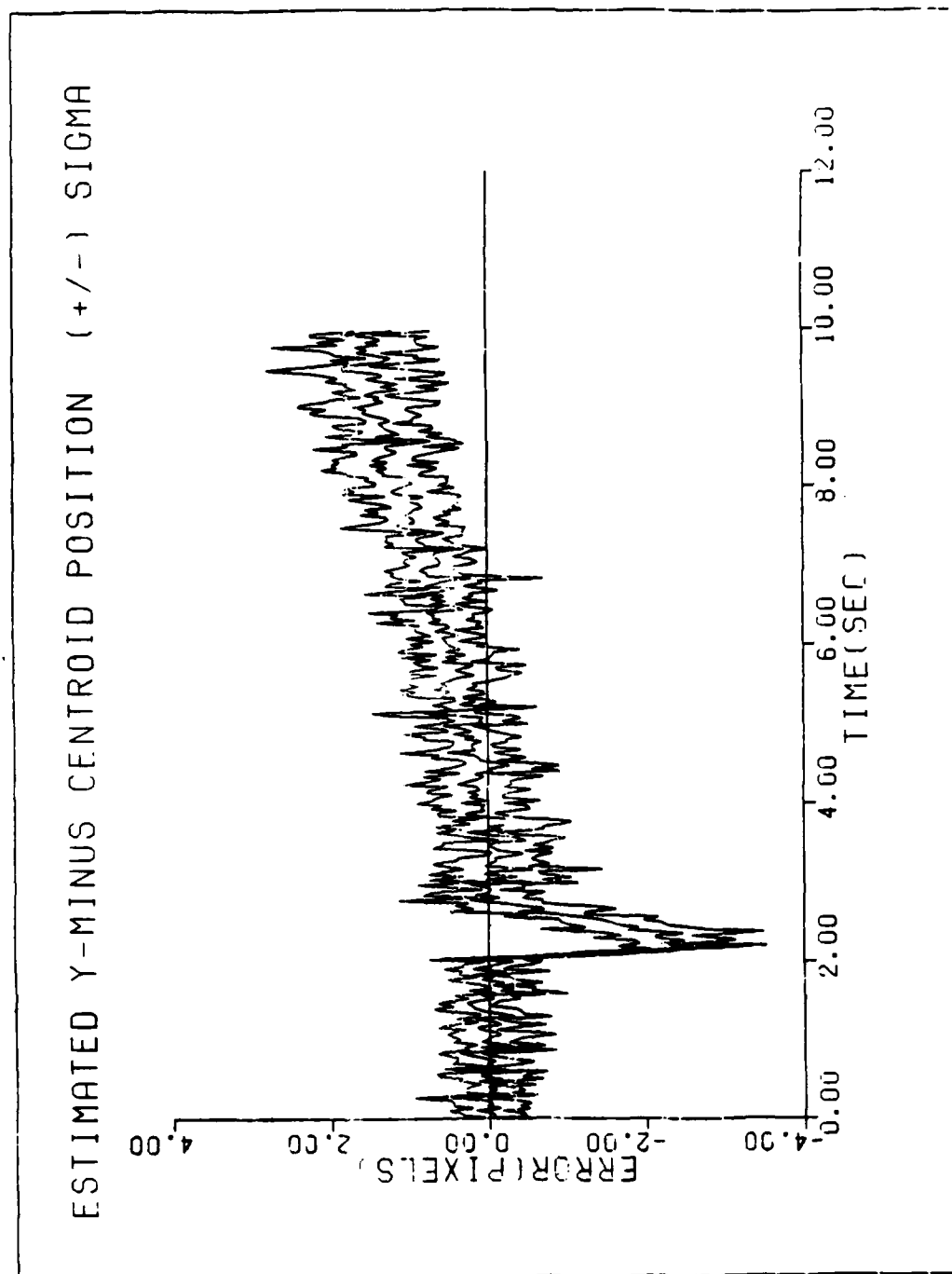


Figure A34 Performance plot for T2G20N

APPENDIX B
NON-IDEAL CONTROLLER INVESTIGATION

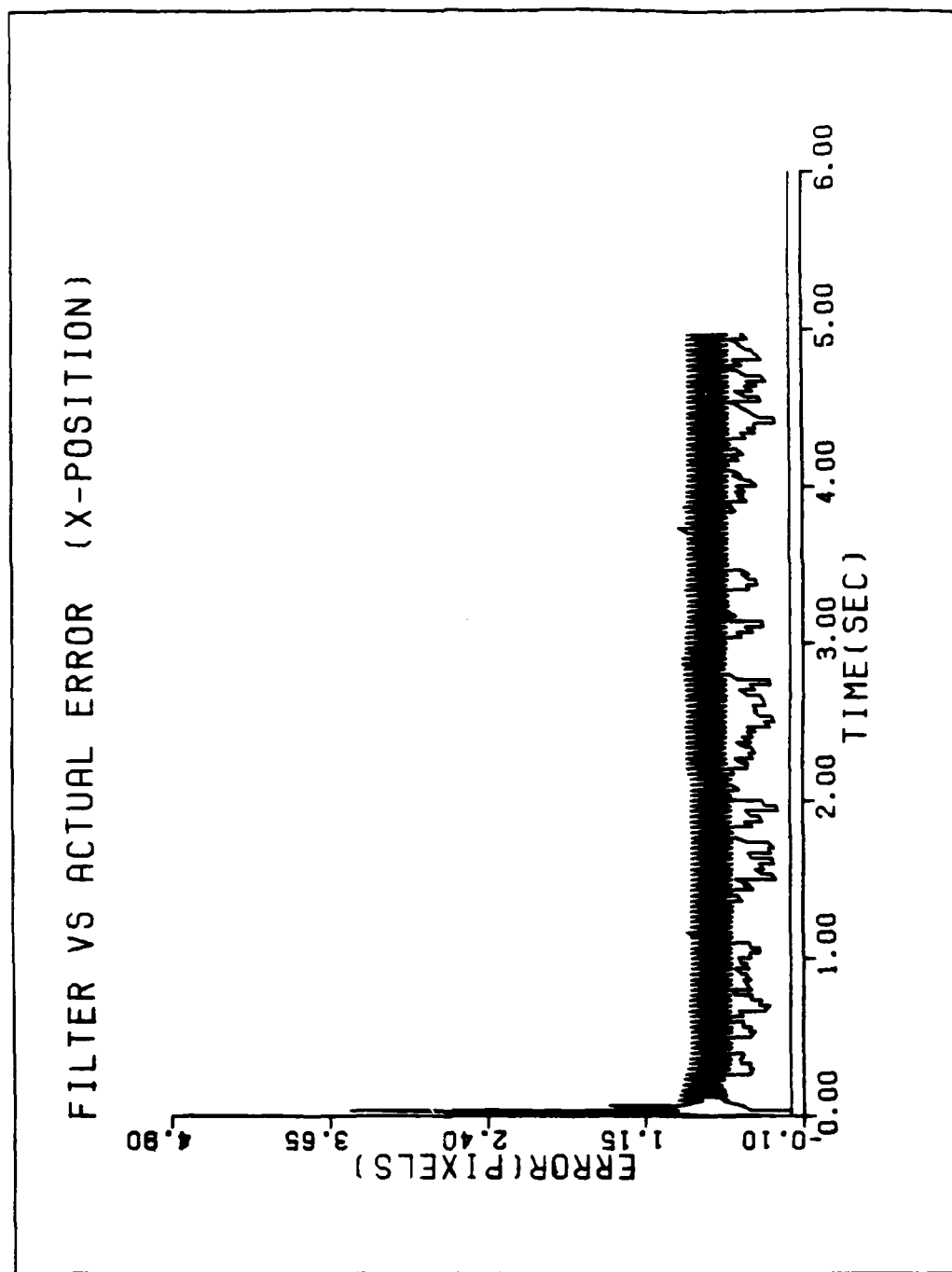


Figure B1 Performance plot for T2G10C.998

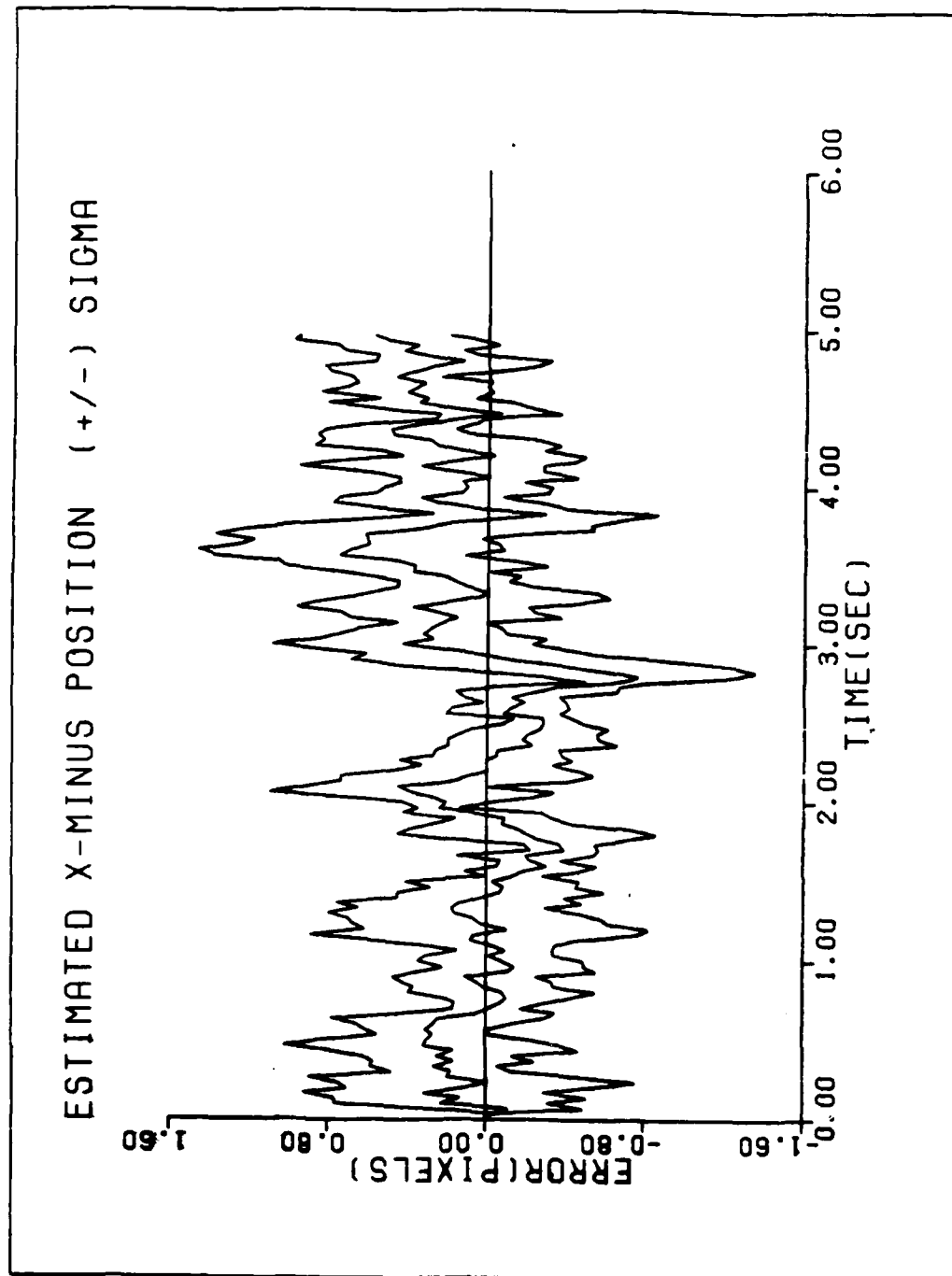


Figure B2 Performance plot for T2G10C.998

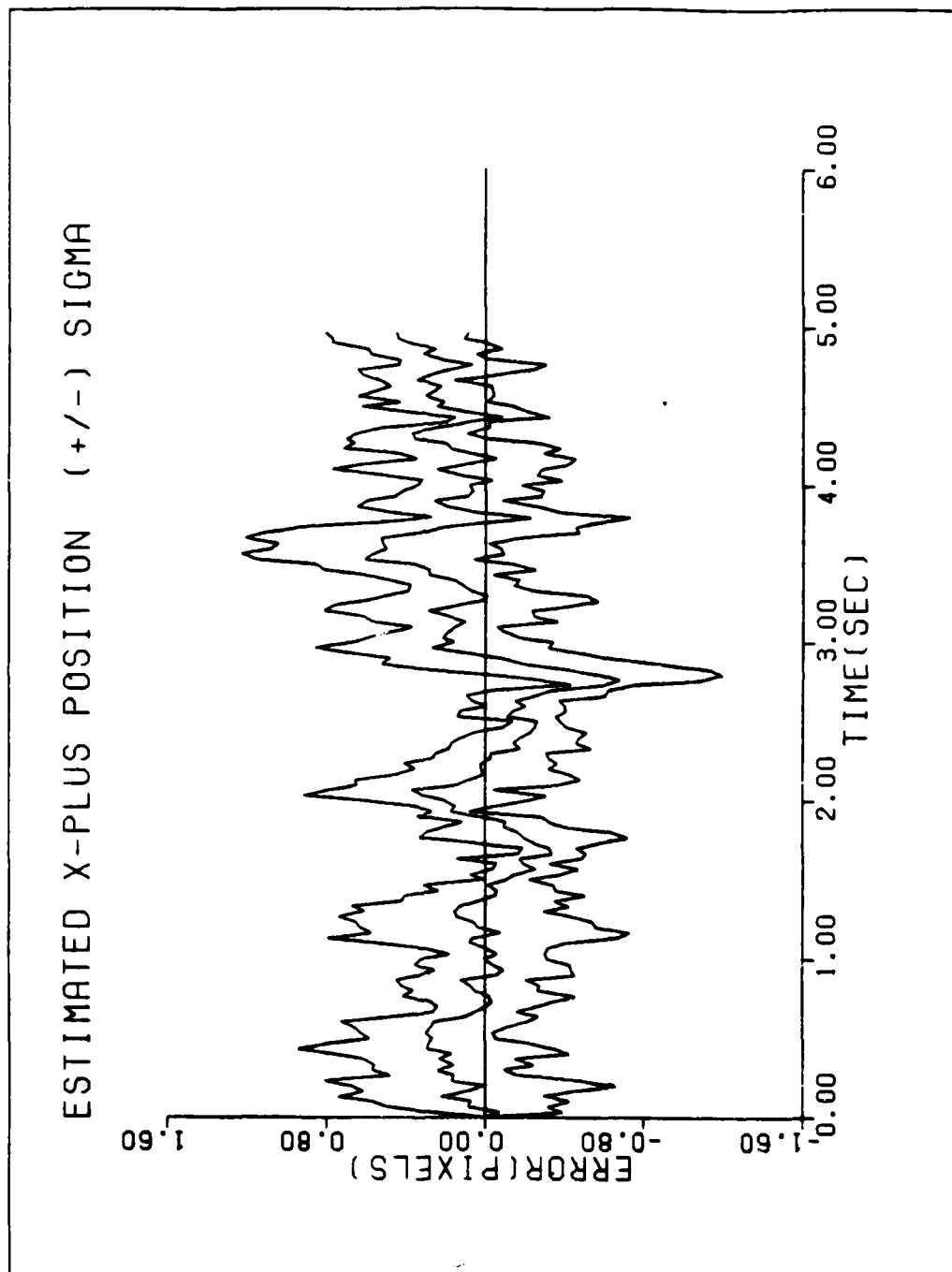


Figure B3 Performance plot for T2G10C.998

CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE
ESTIMATION FOR TRACKING.. (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. A S NETZER
DEC 85 AFIT/GAE/ENG/85D-2 F/G 17/5

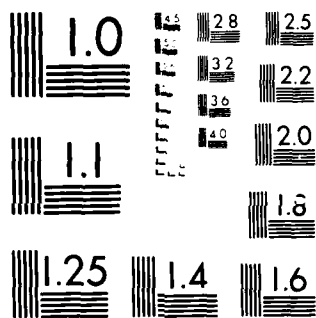
組

UNCLASSIFIED

DEC 85 AFIT/GAE/ENG/85D-2

F/G 17/5

組



MICROCOPY RESOLUTION TEST CHART
 NATIONAL BUREAU OF STANDARDS-1963-A

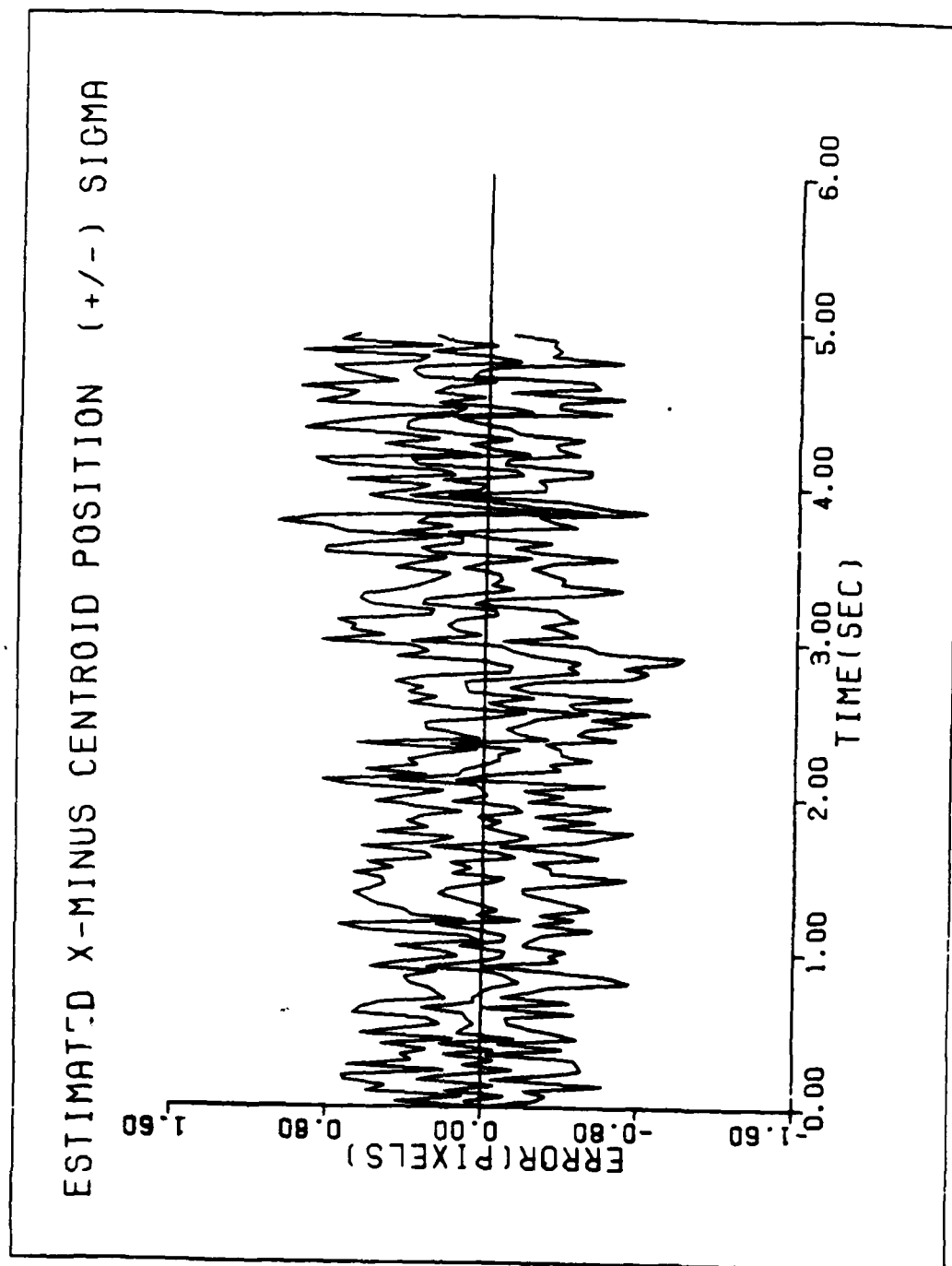


Figure B4 Performance plot for T2G10C.998

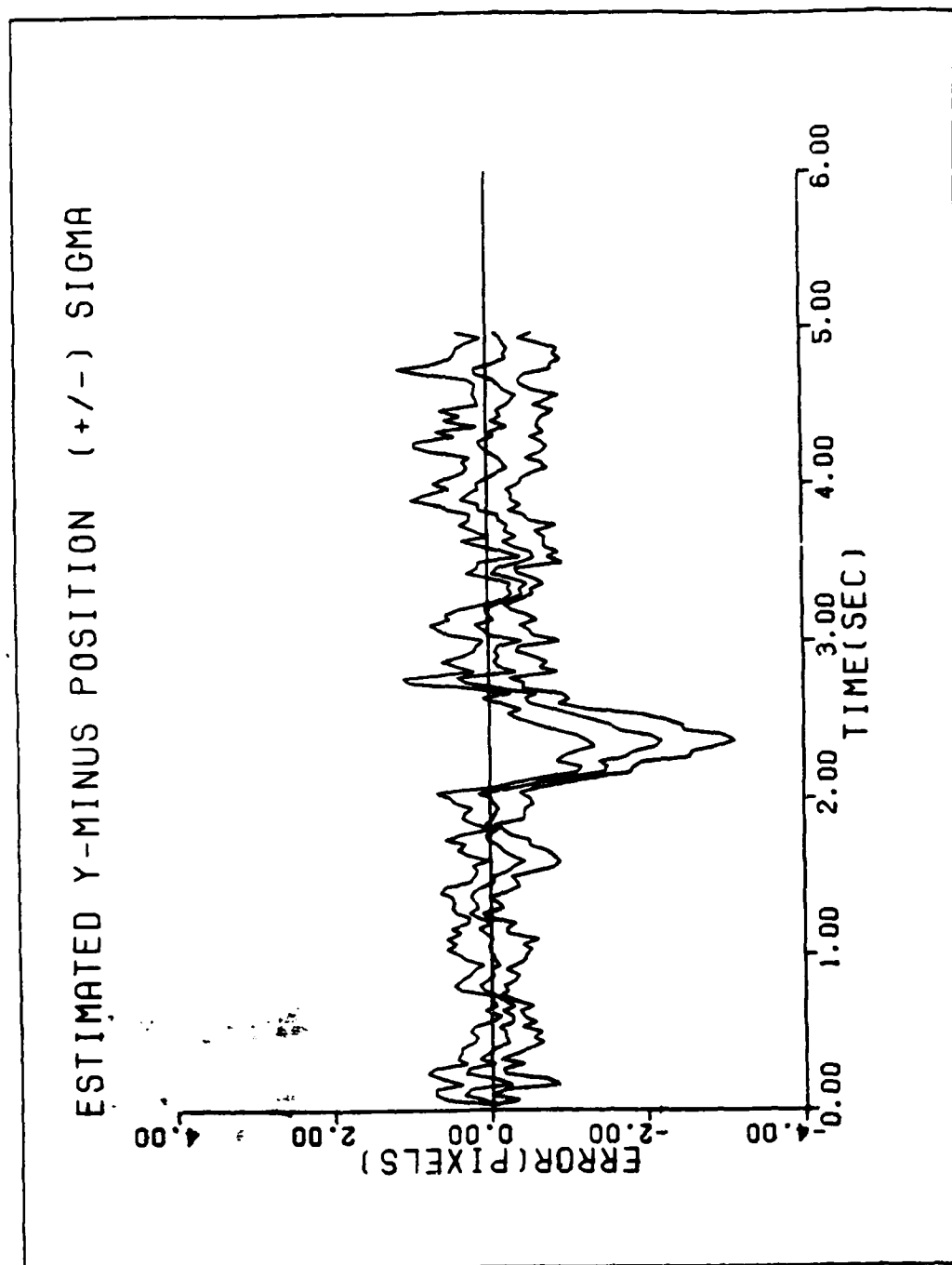


Figure B5 Performance plot for T2G10C.998

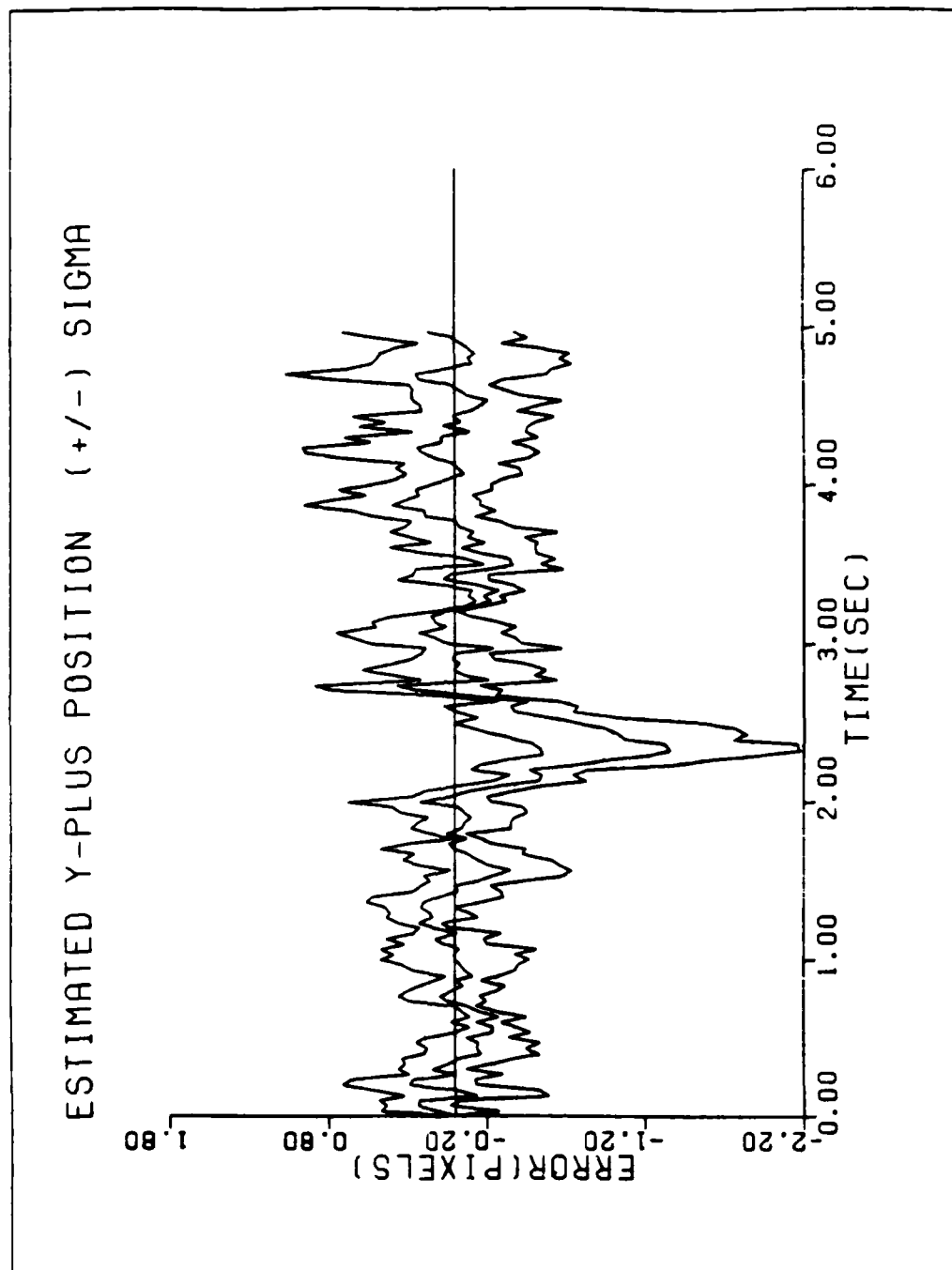


Figure B6 Performance plot for T2G10C.998

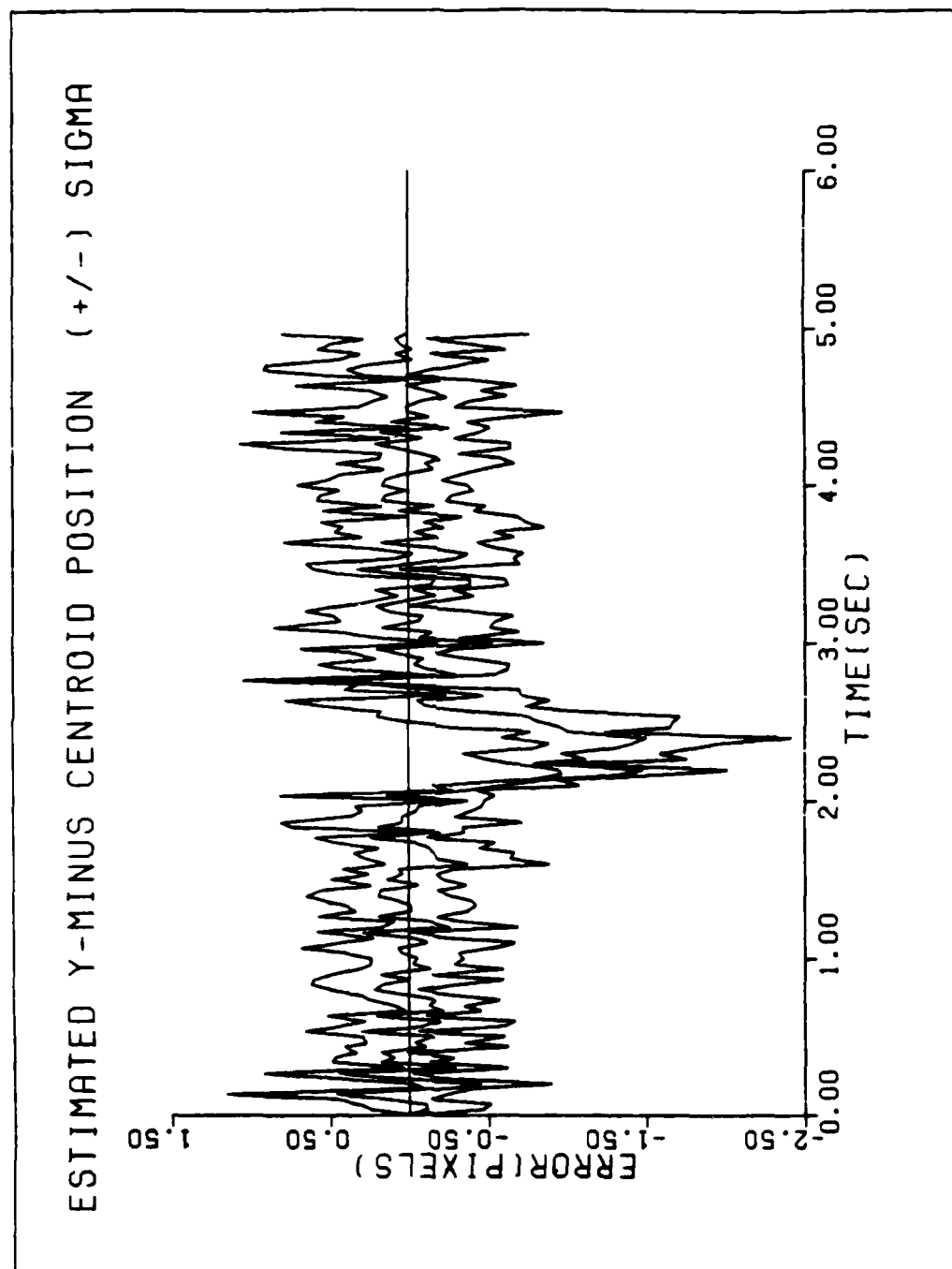


Figure B7 Performance plot for T2G10C.998

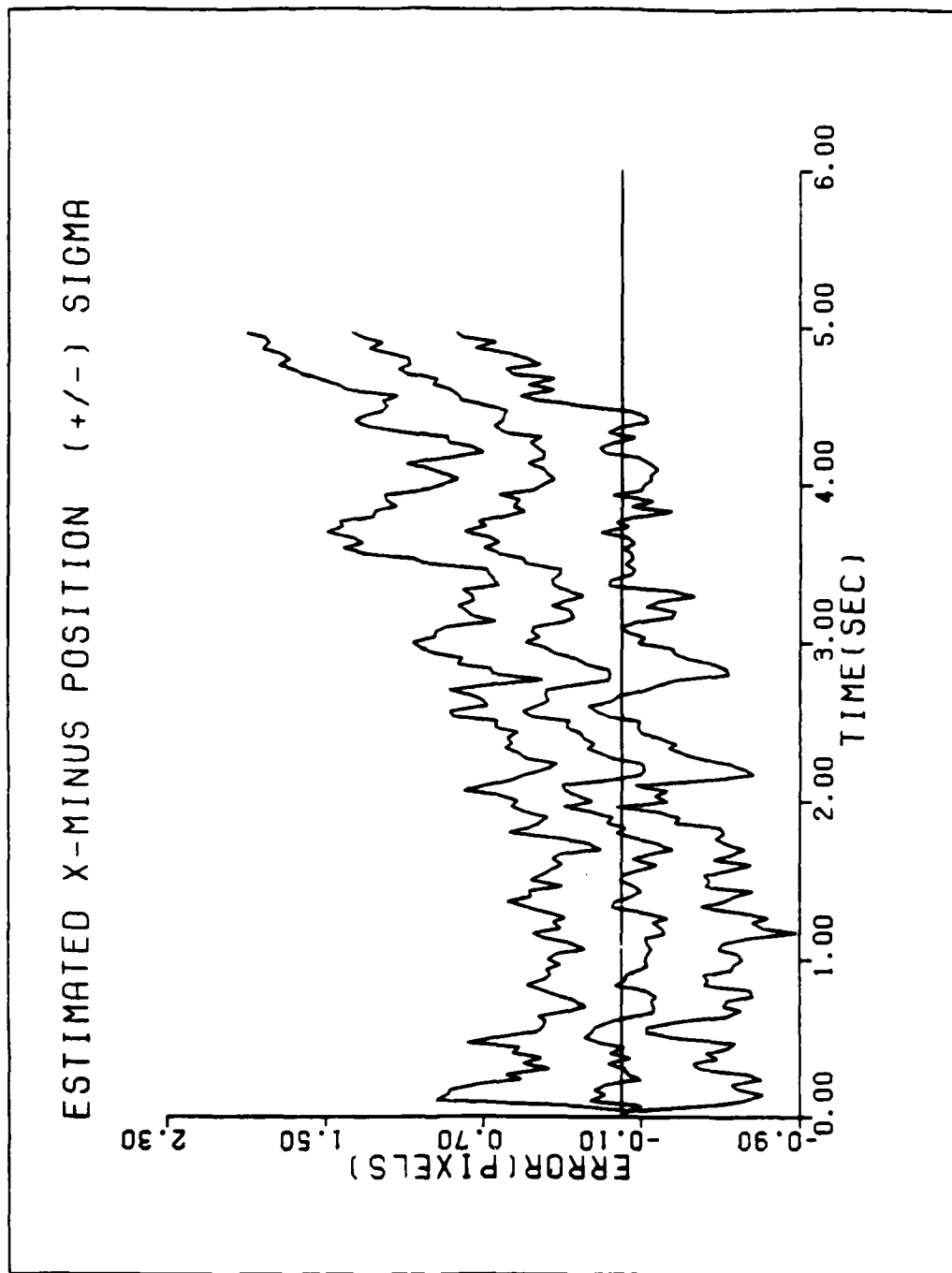


Figure B8 Performance plot for T2G10C.967

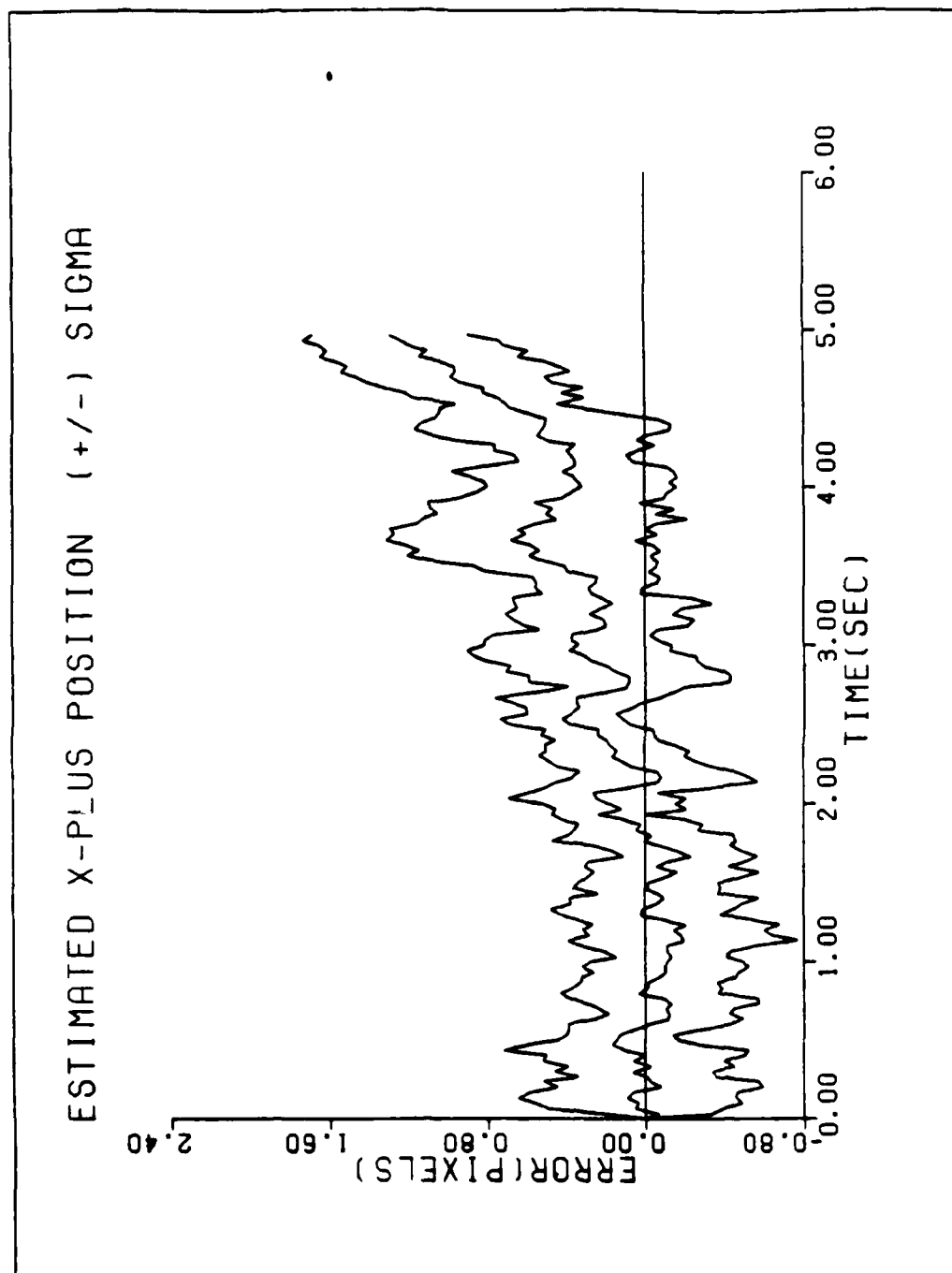


Figure B9 Performance plot for T2G10C.967

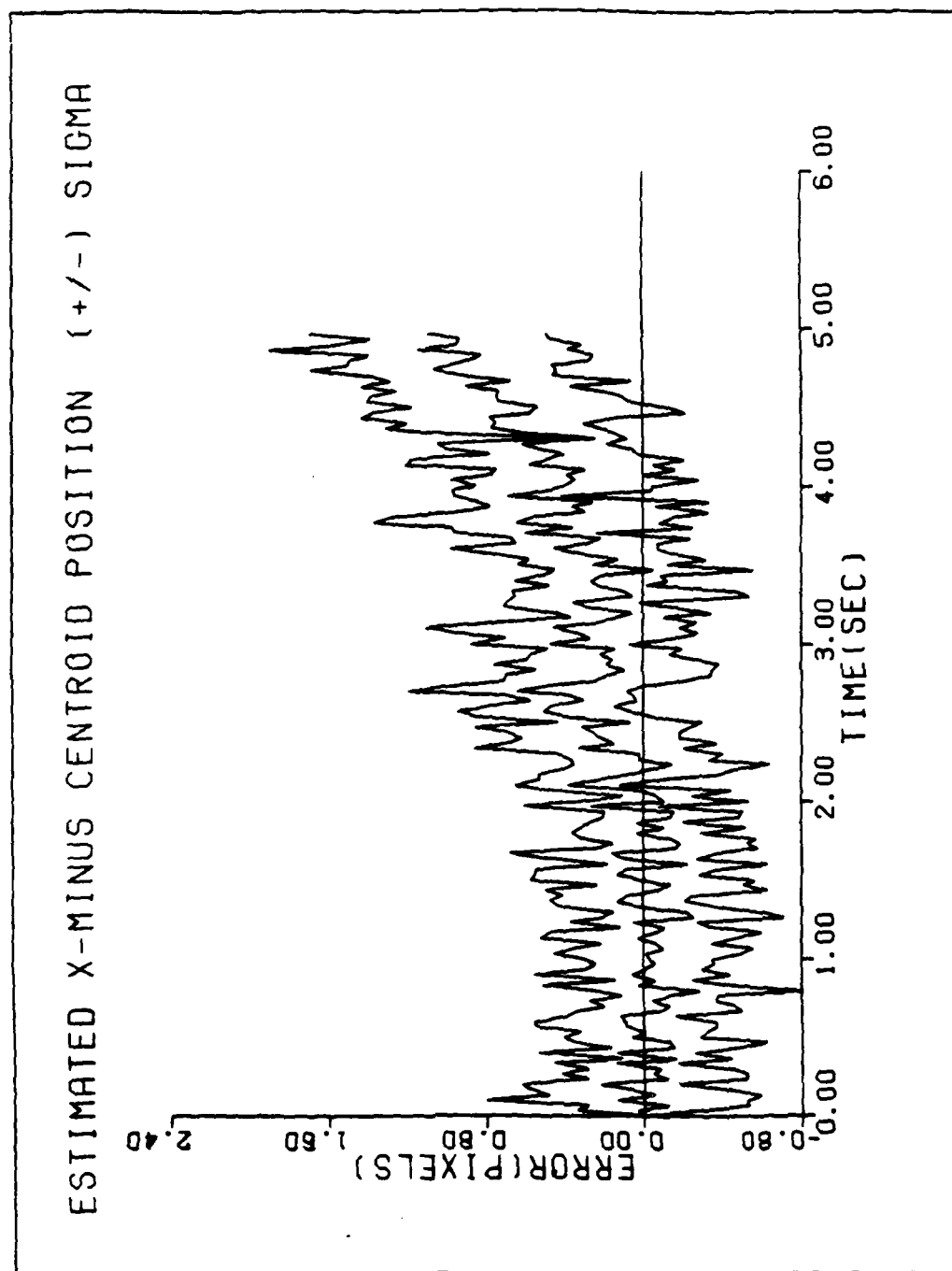


Figure B10 Performance plot for T2G10C.967

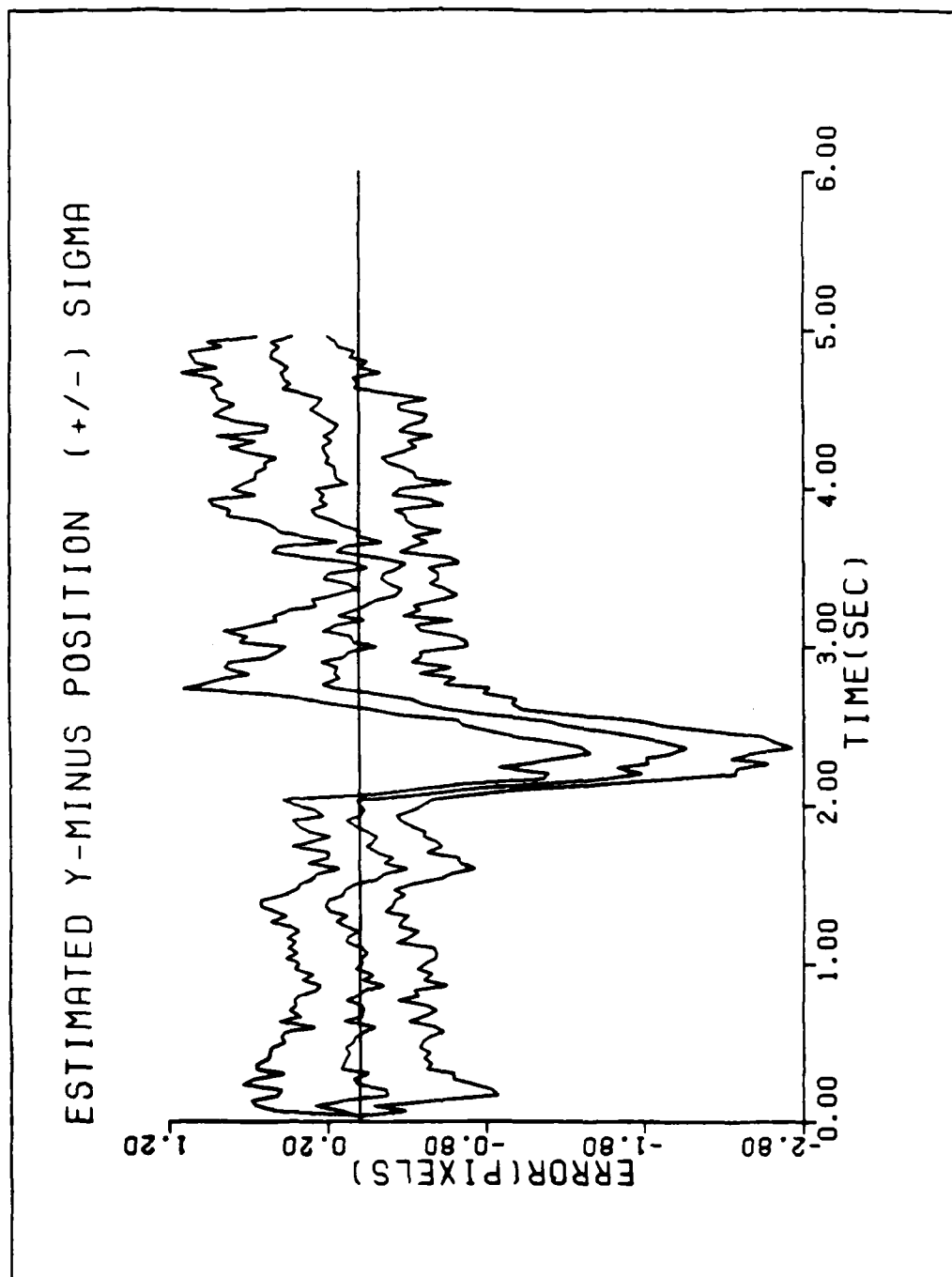


Figure B11 Performance plot for T2G10C.967

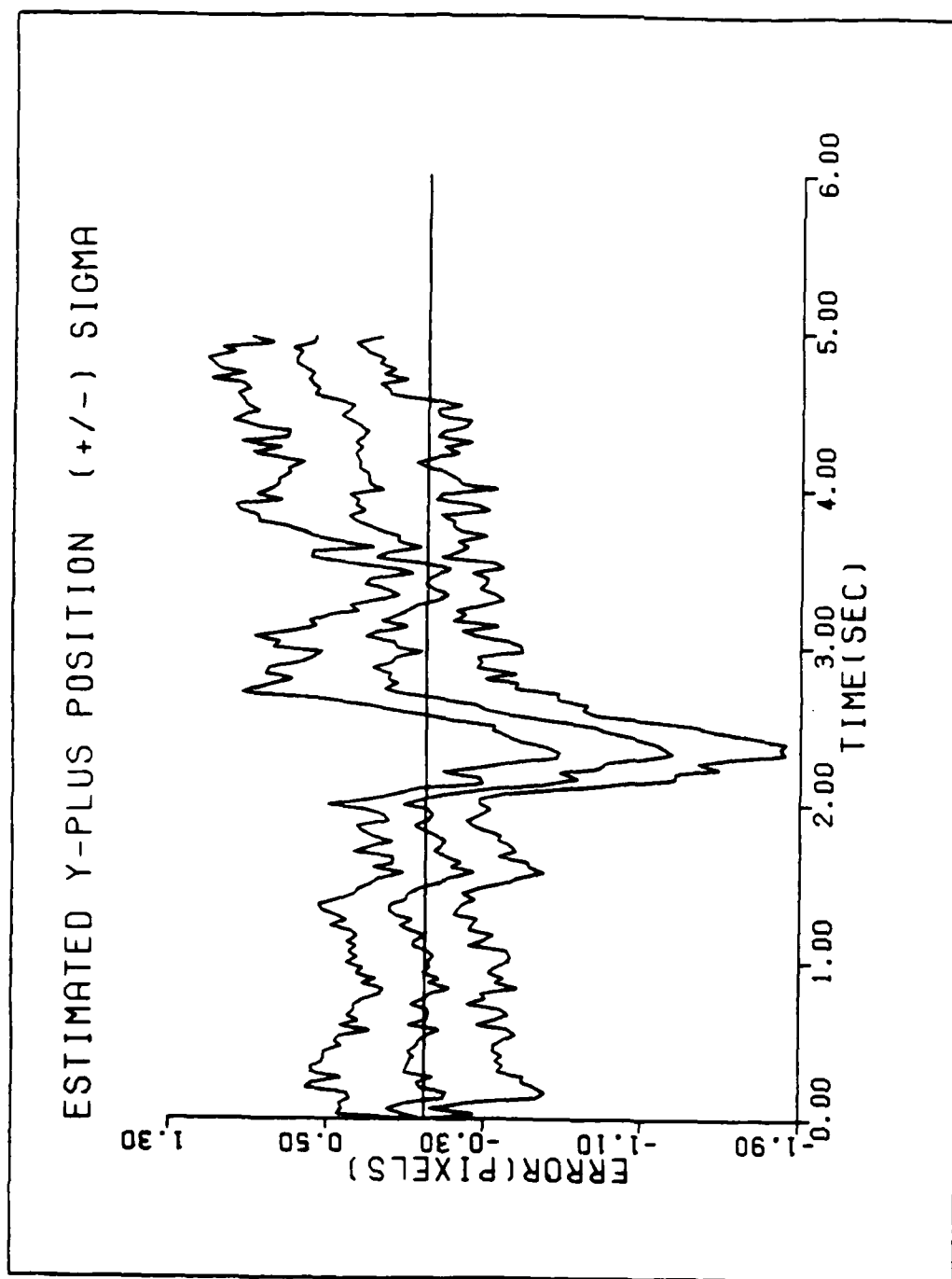


Figure B12 Performance plot for T2G10C.967

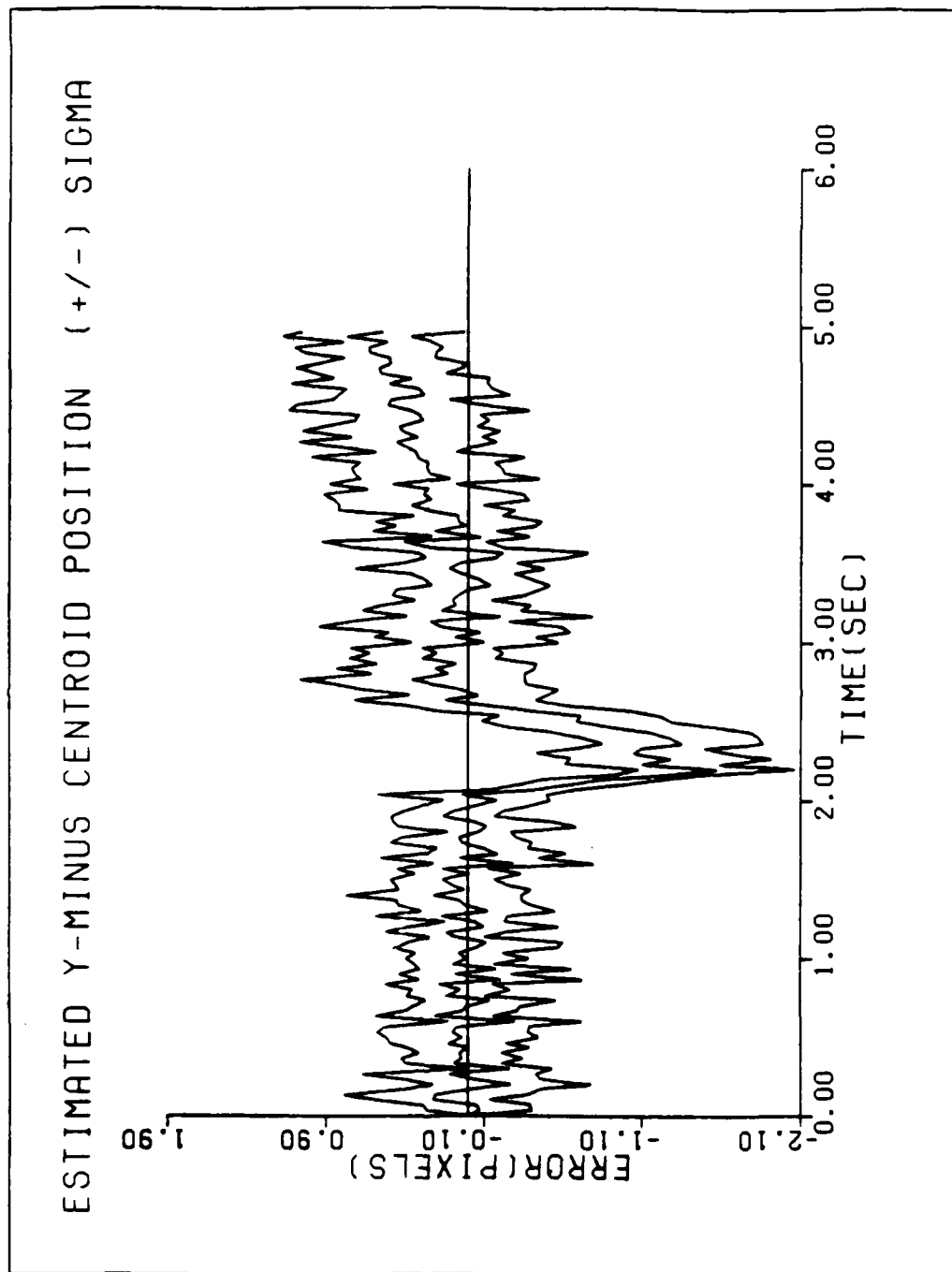


Figure B13 Performance plot for T2G10C.967

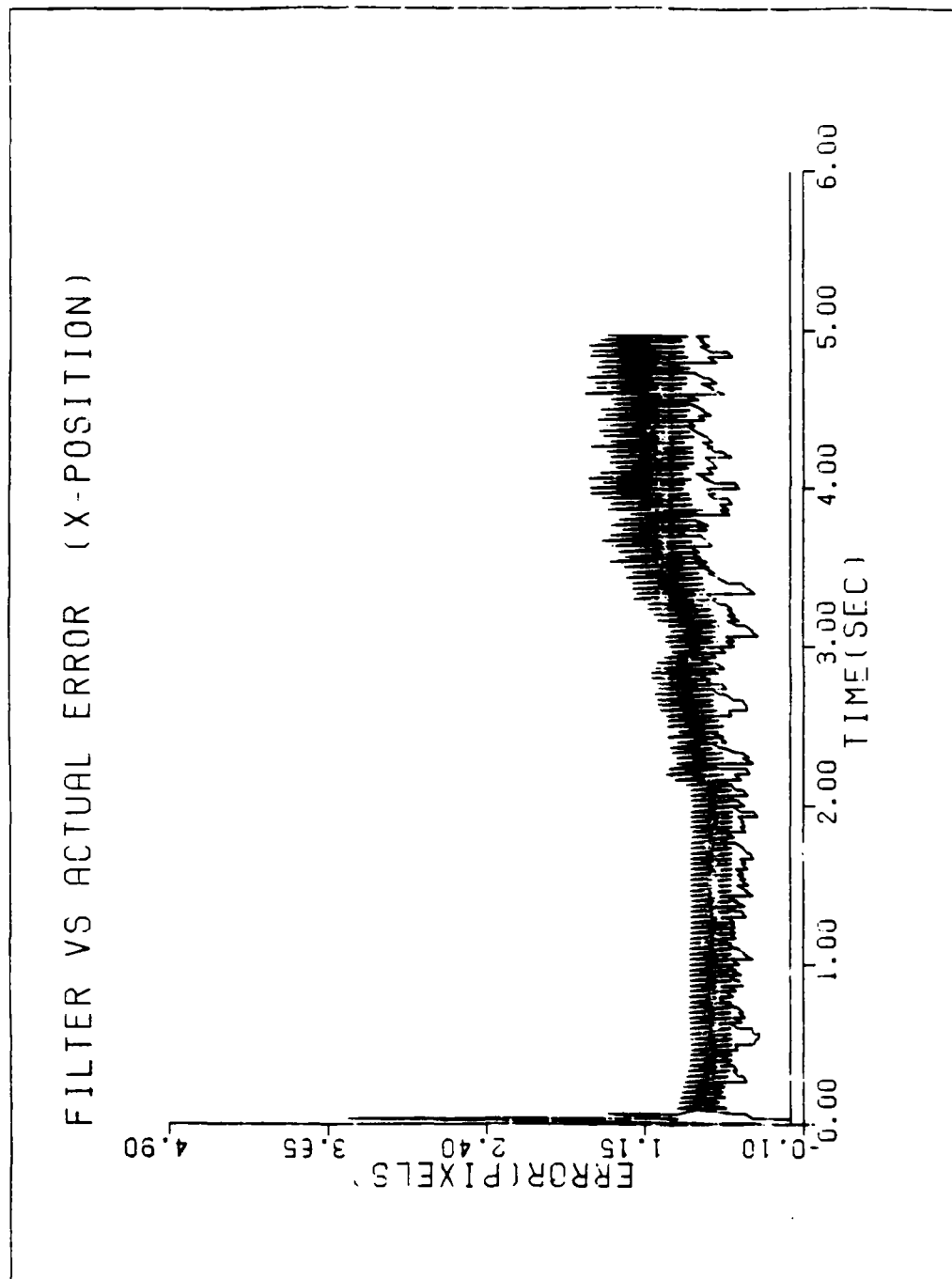


Figure B14 Performance plot for T2G10C.95

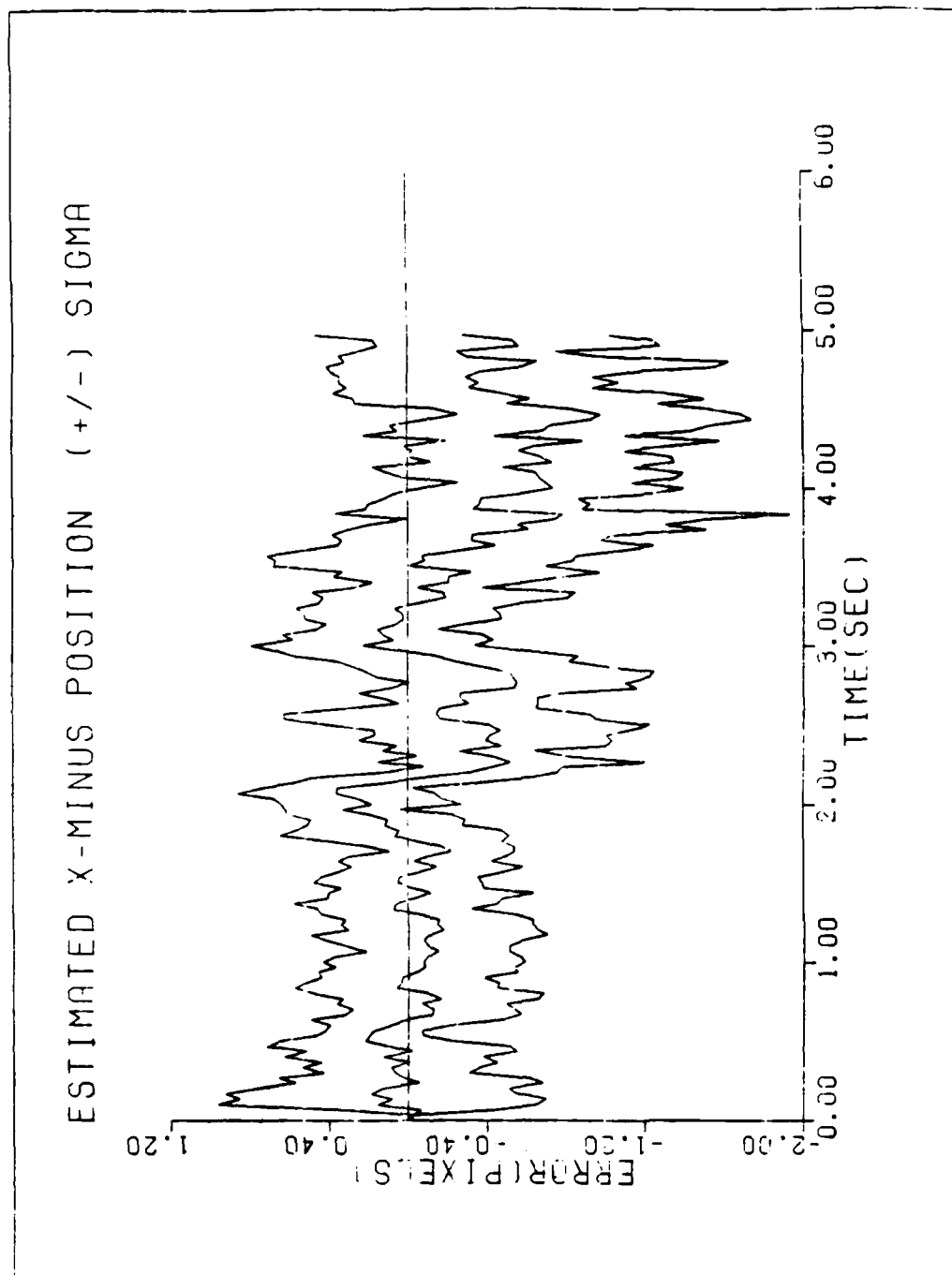


Figure B15 Performance plot for T2G10C.95

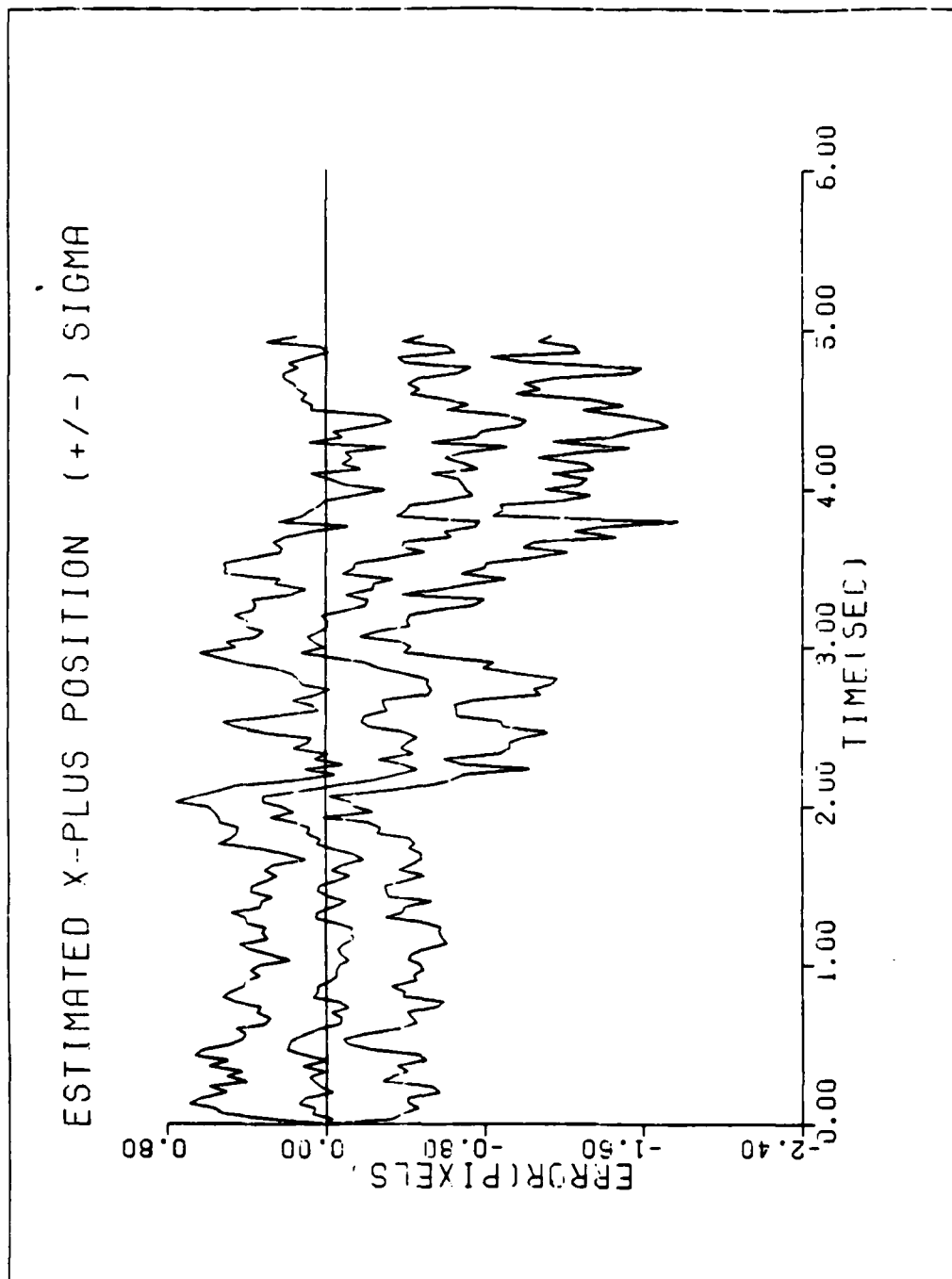


Figure B16 Performance plot for T2G10C.95

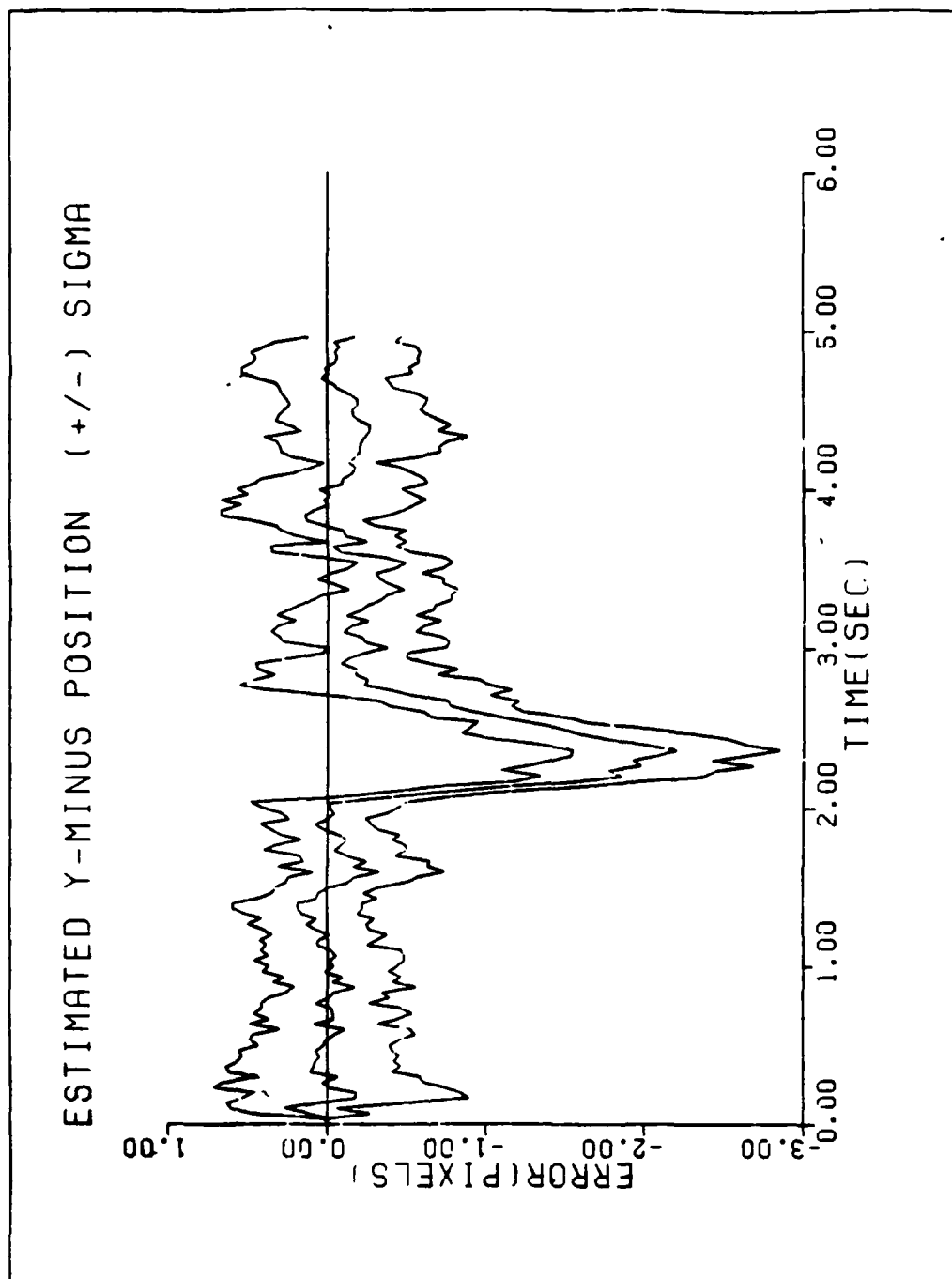


Figure B17 Performance plot for T2G10C.95

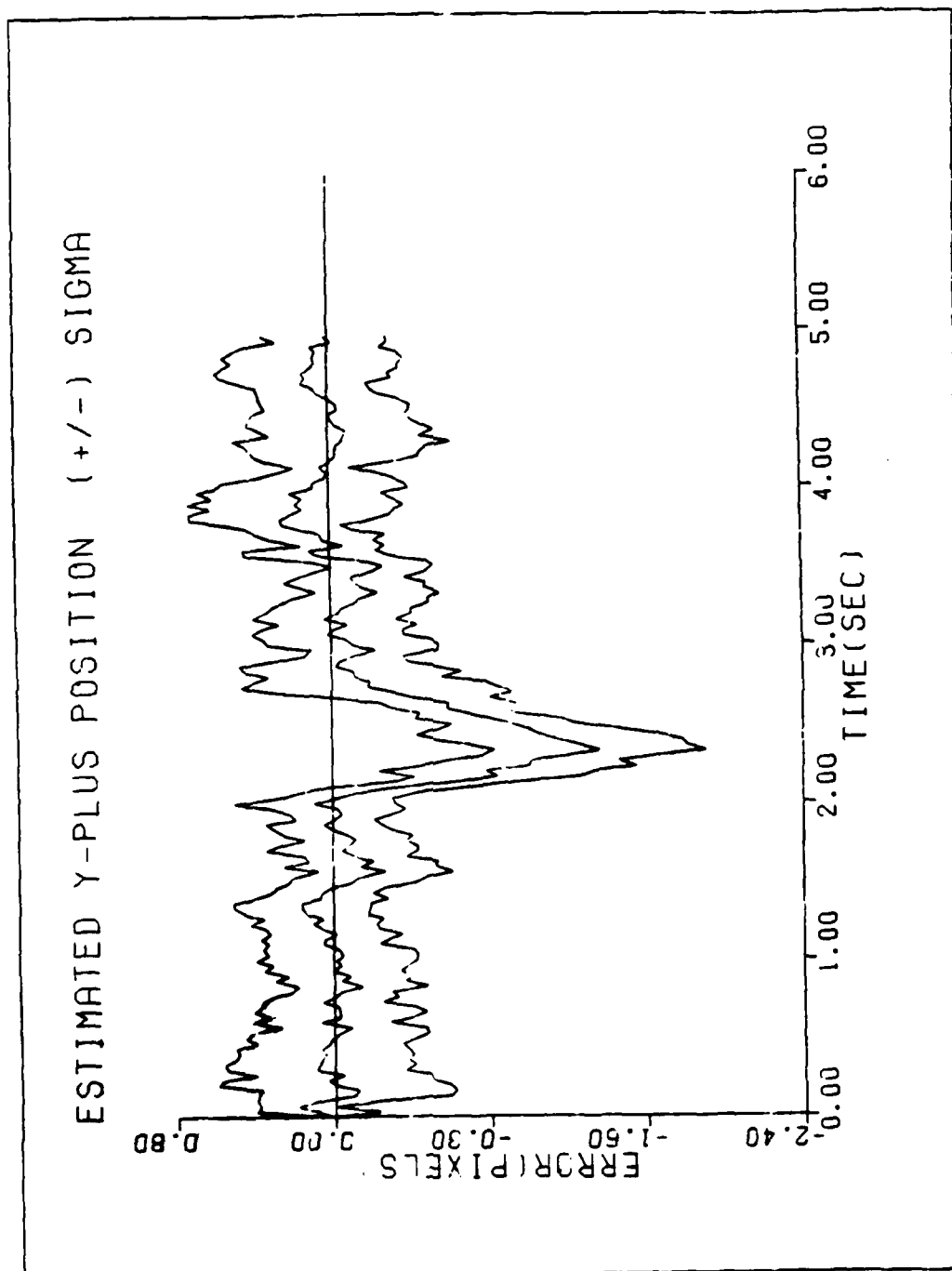


Figure B18 Performance plot for T2G10C.95

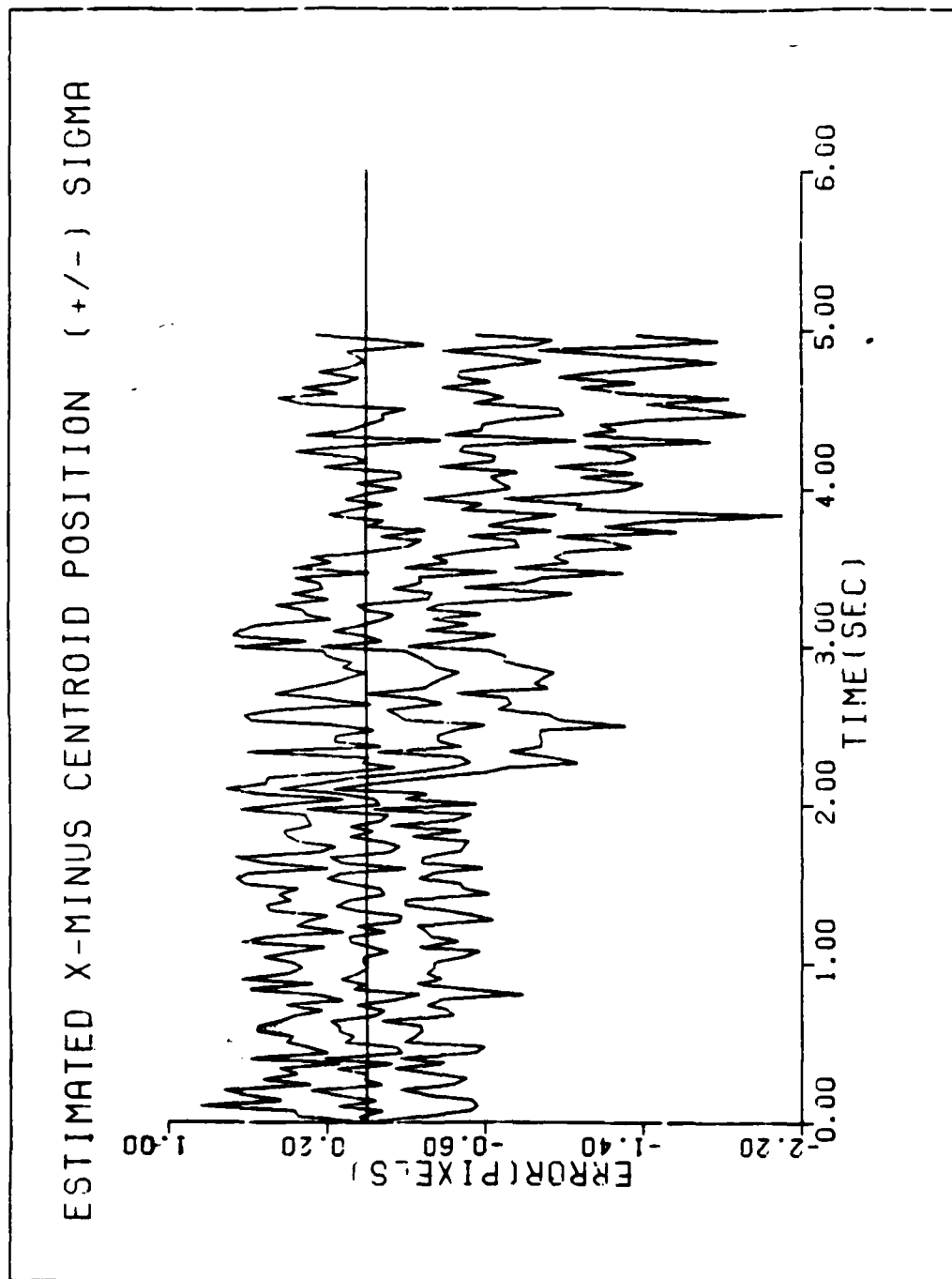


Figure B19 Performance plot for T2G10C.95

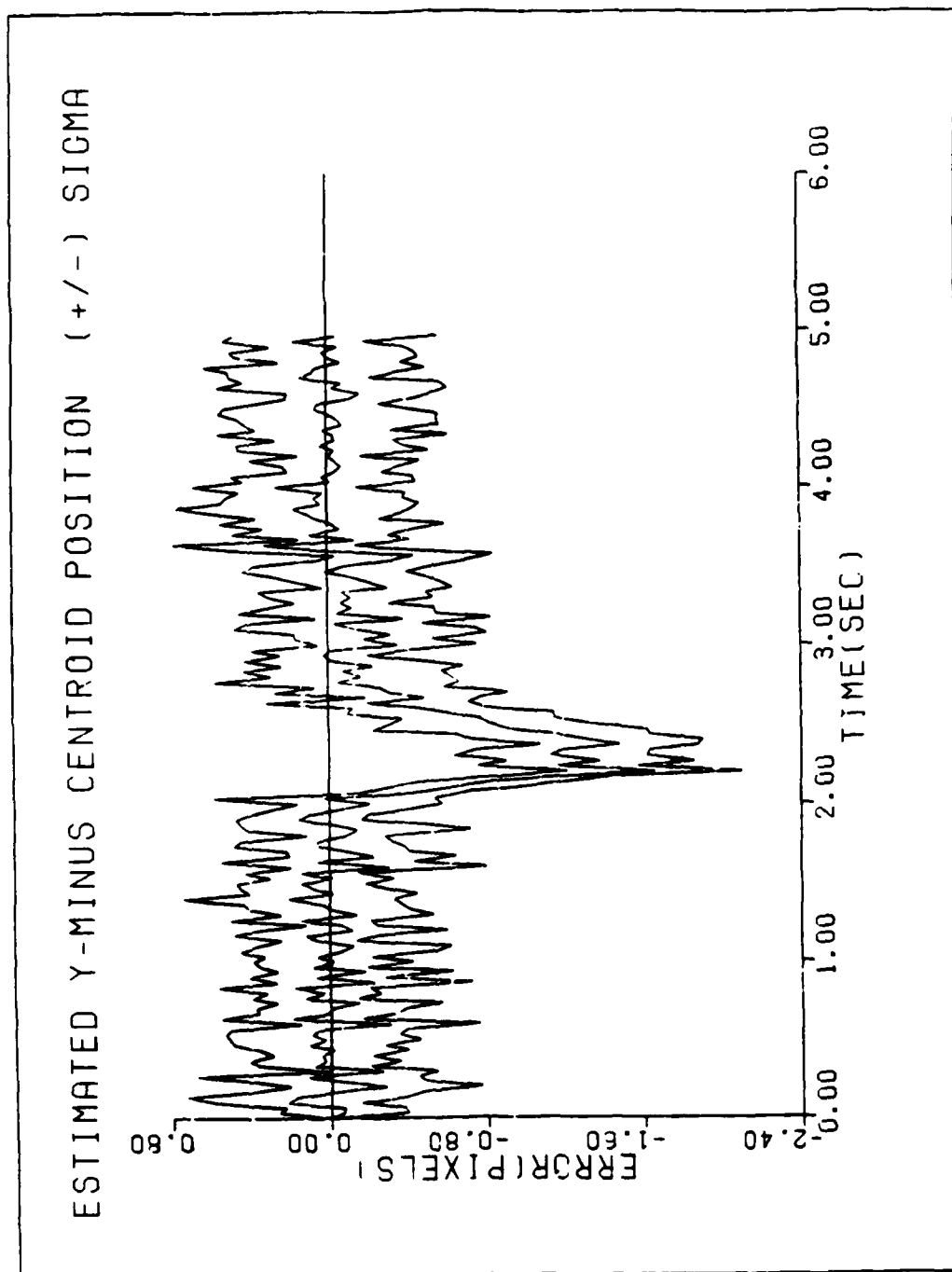


Figure B20 Performance plot for T2G10C.95

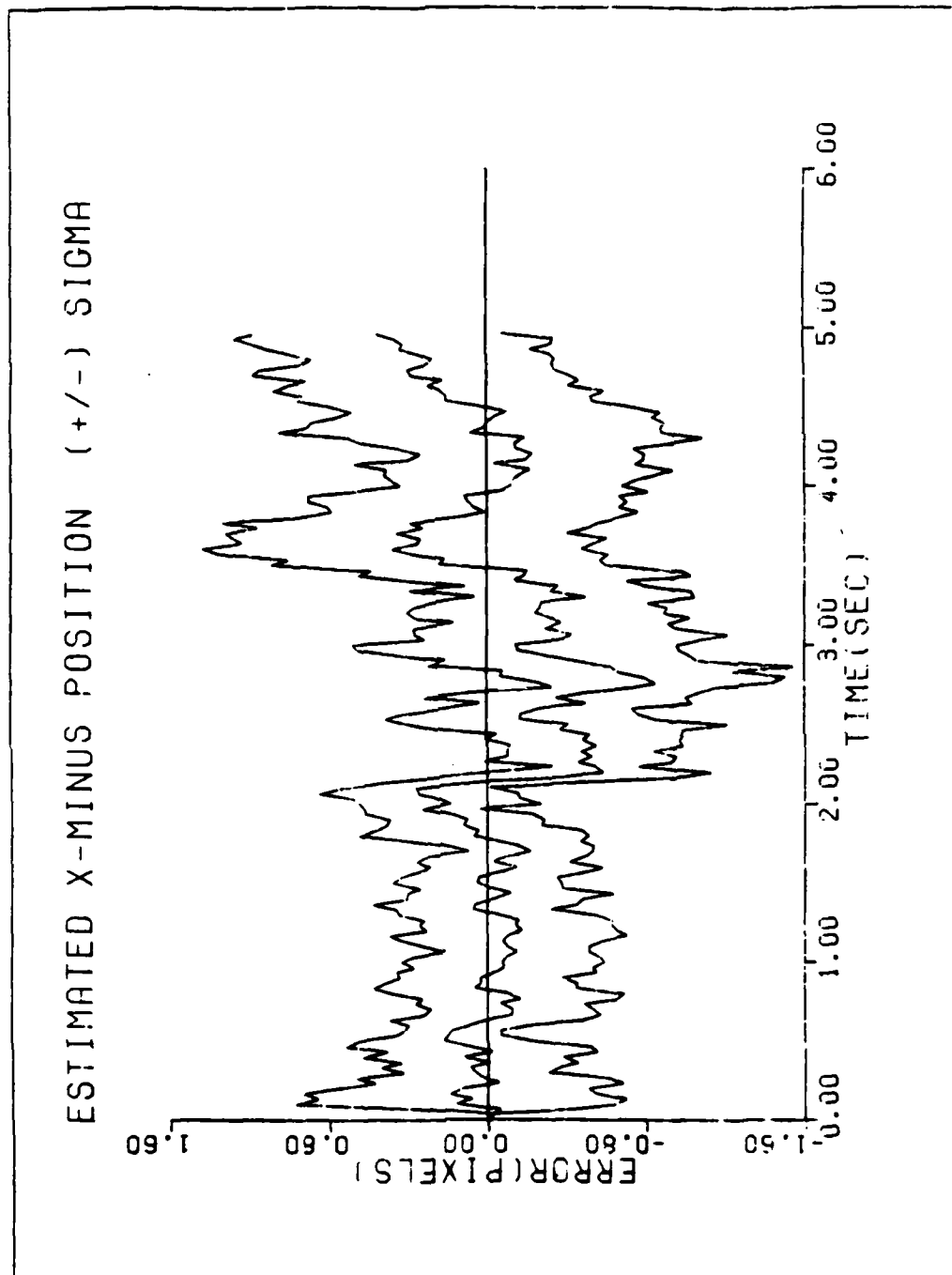


Figure B21 Performance plot for T2G20C.95

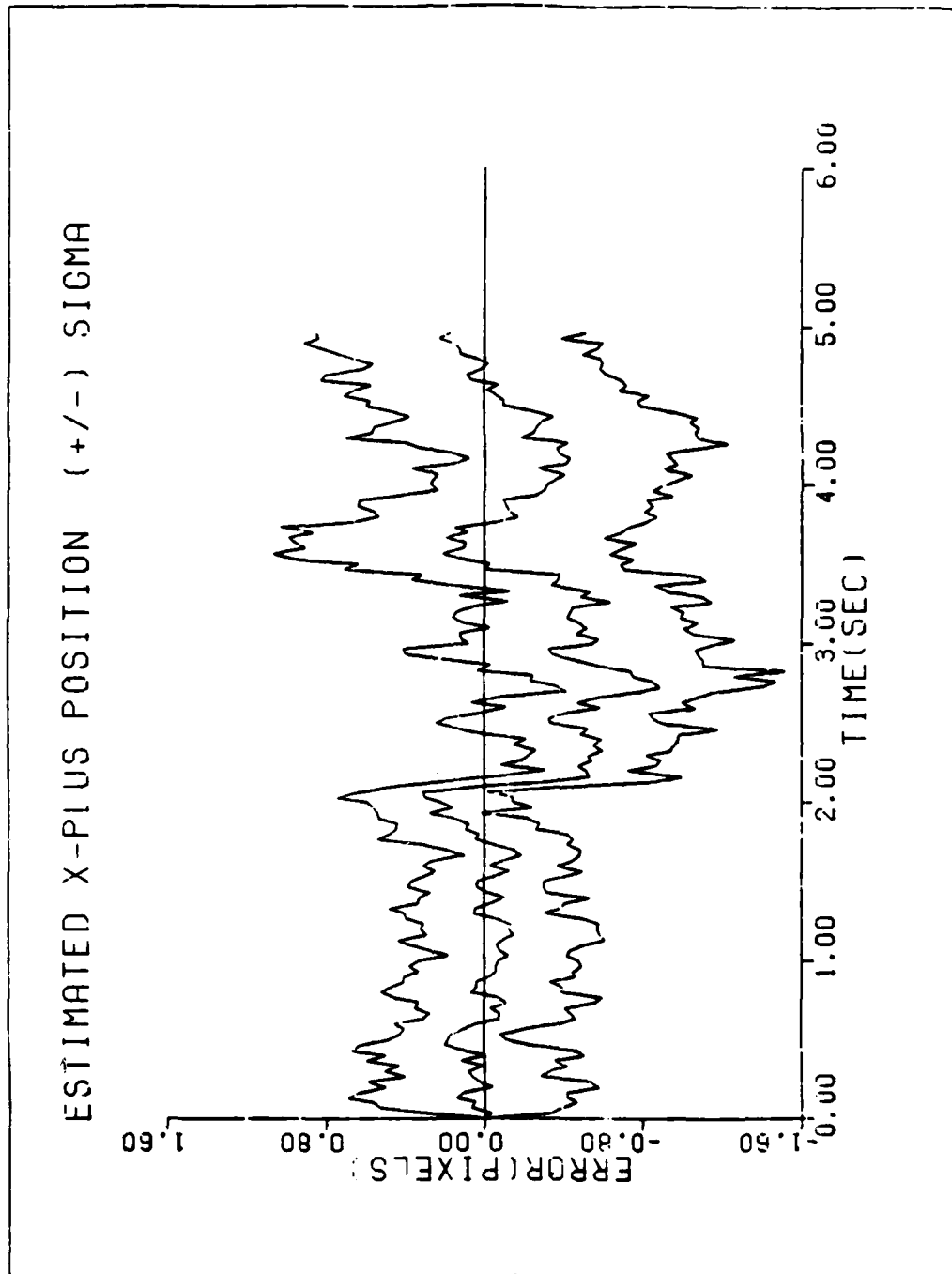


Figure B22 Performance plot for T2G20C.95

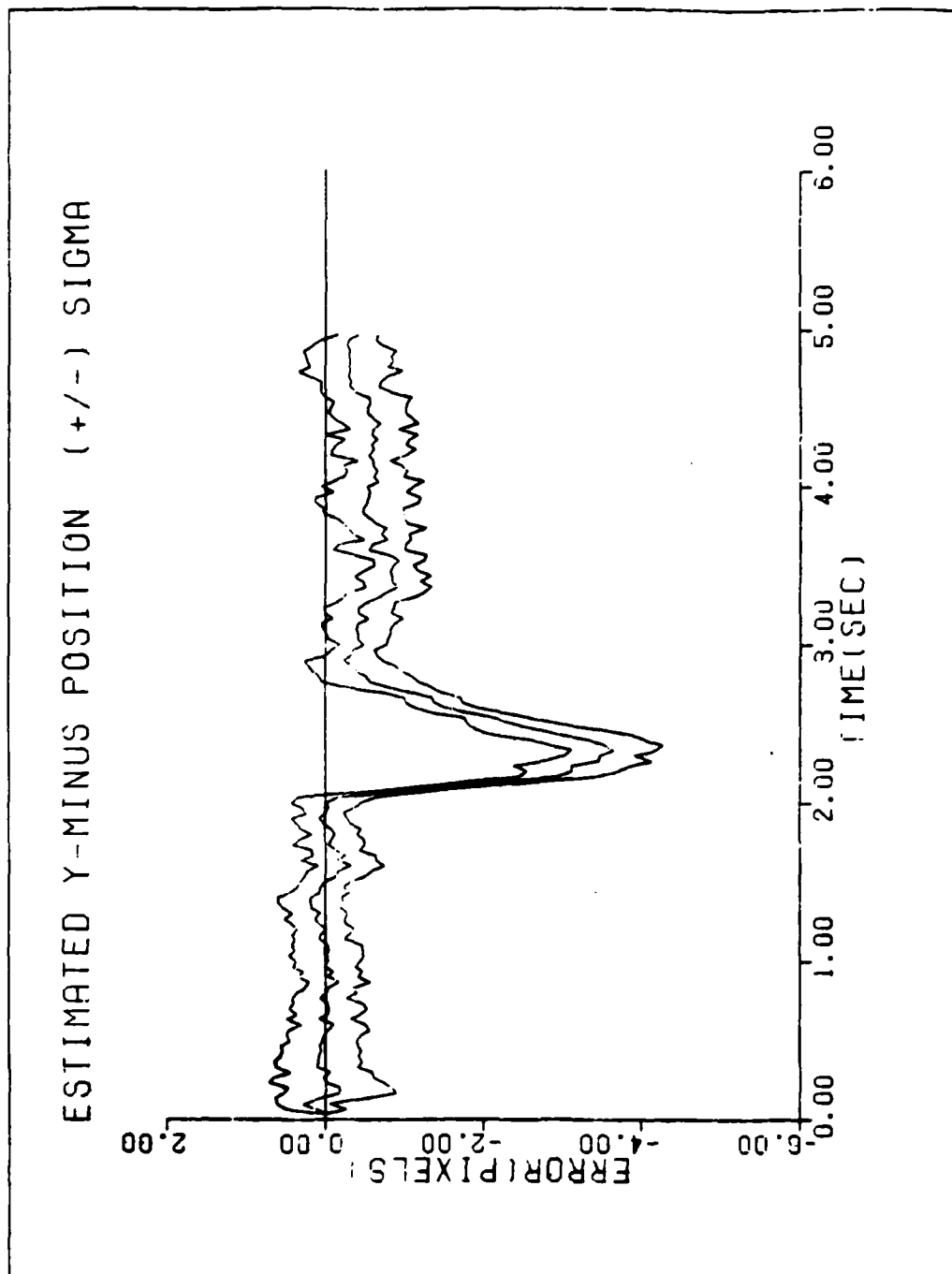


Figure B23 Performance plot for T2G20C.95

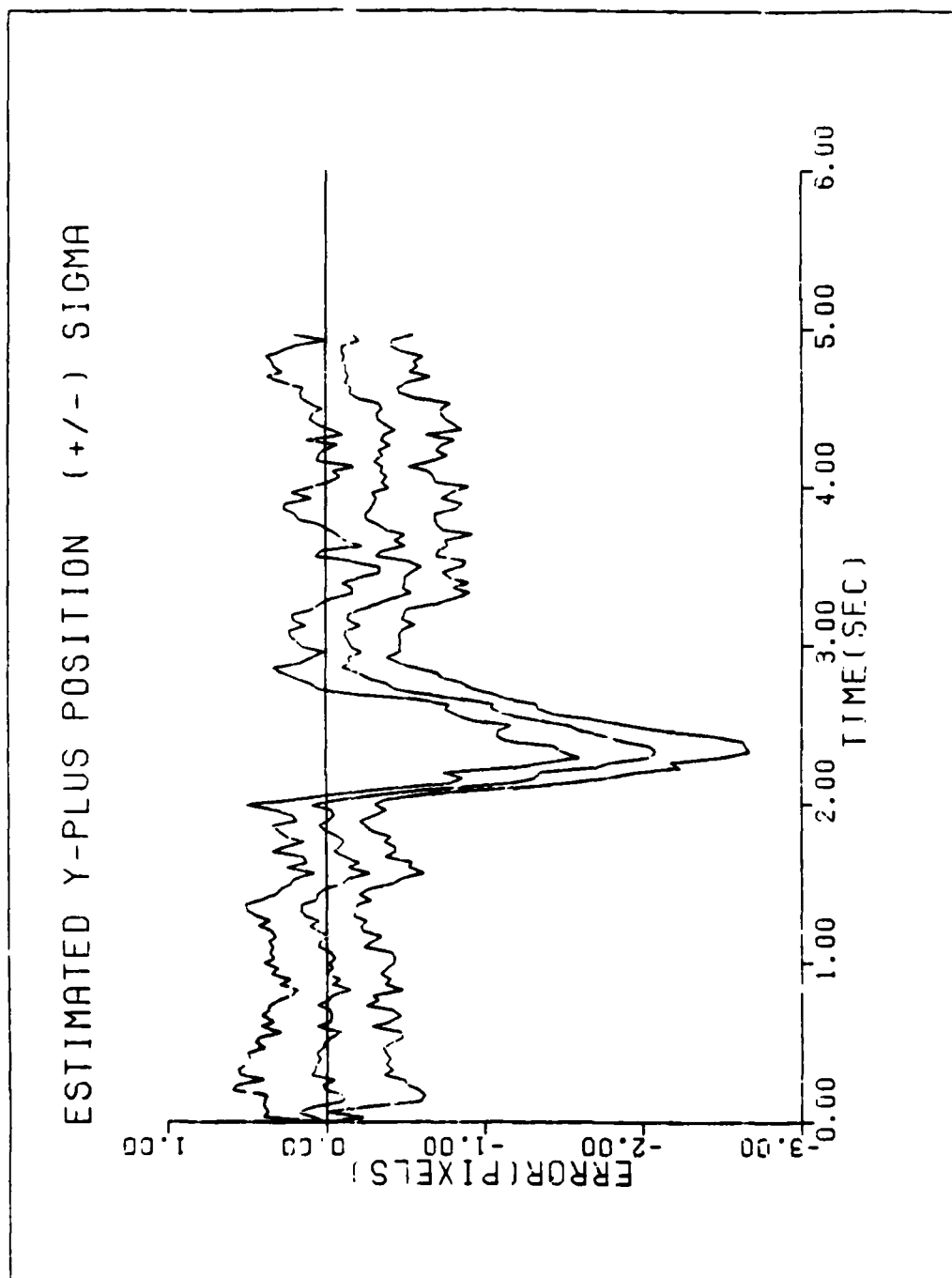


Figure B24 Performance plot for T2G20C.95

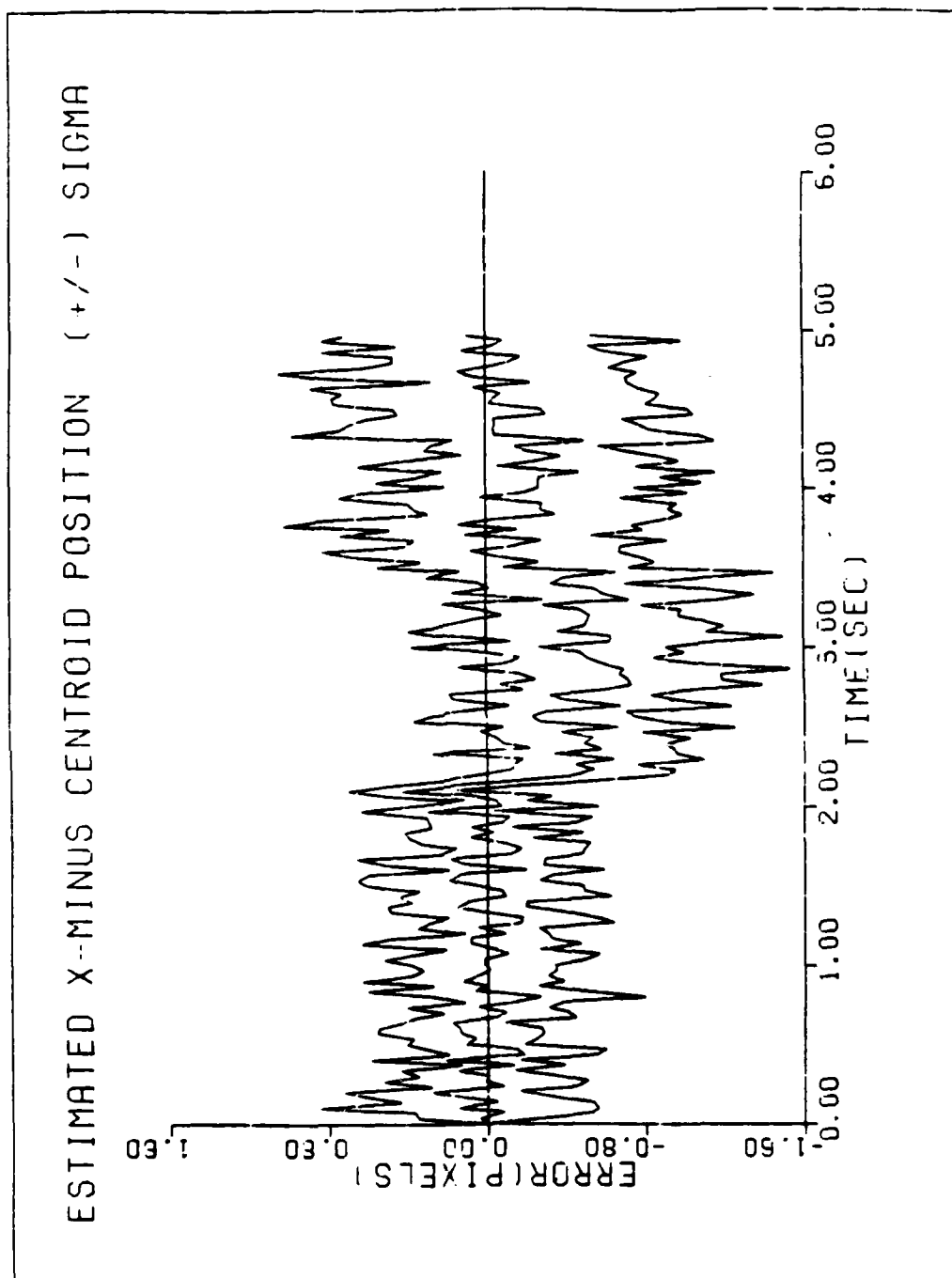


Figure B25 Performance plot for T2G20C.95

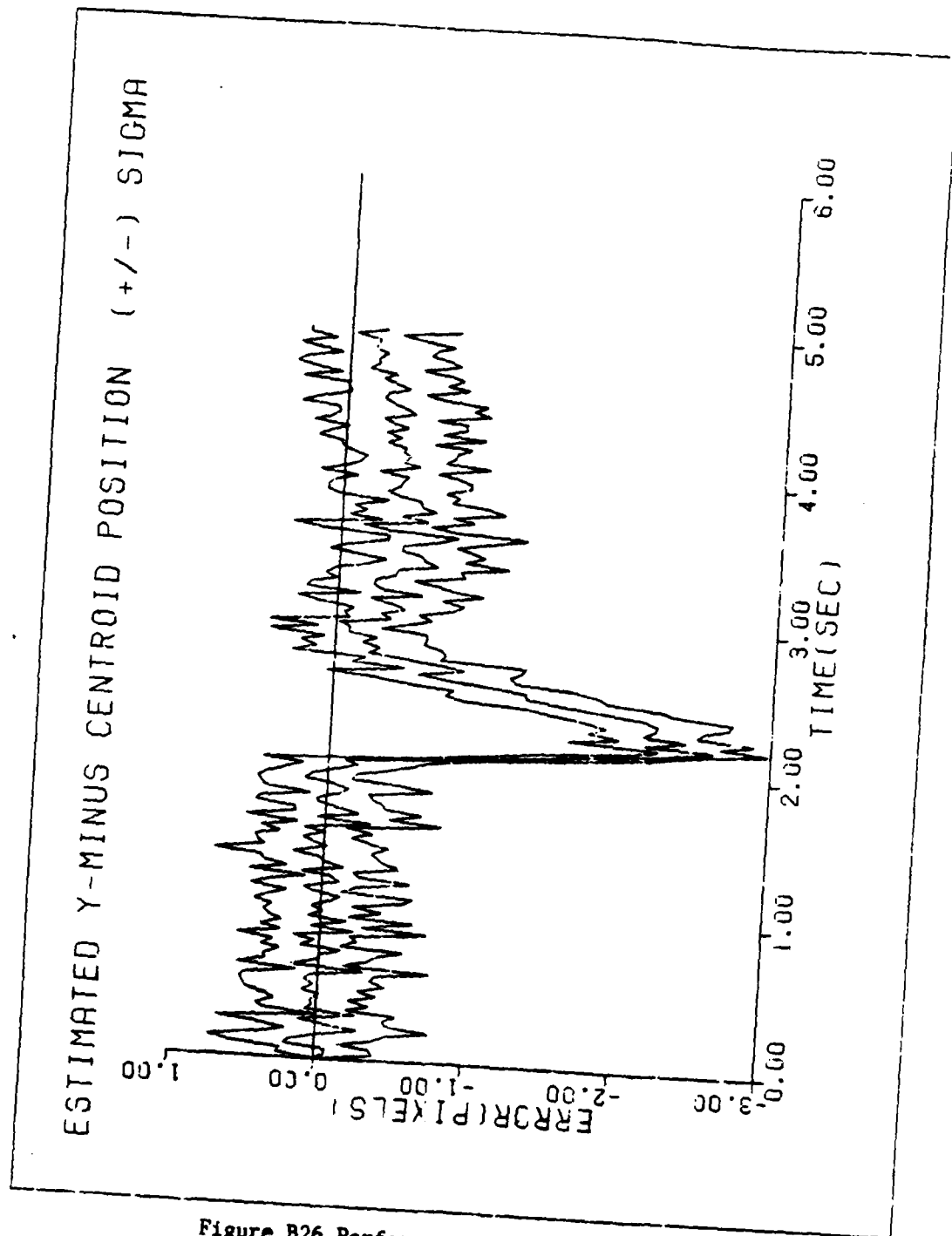


Figure B26 Performance plot for T2G20C.95

APPENDIX C
SIGNAL TO NOISE RATIO SENSITIVITY ANALYSIS

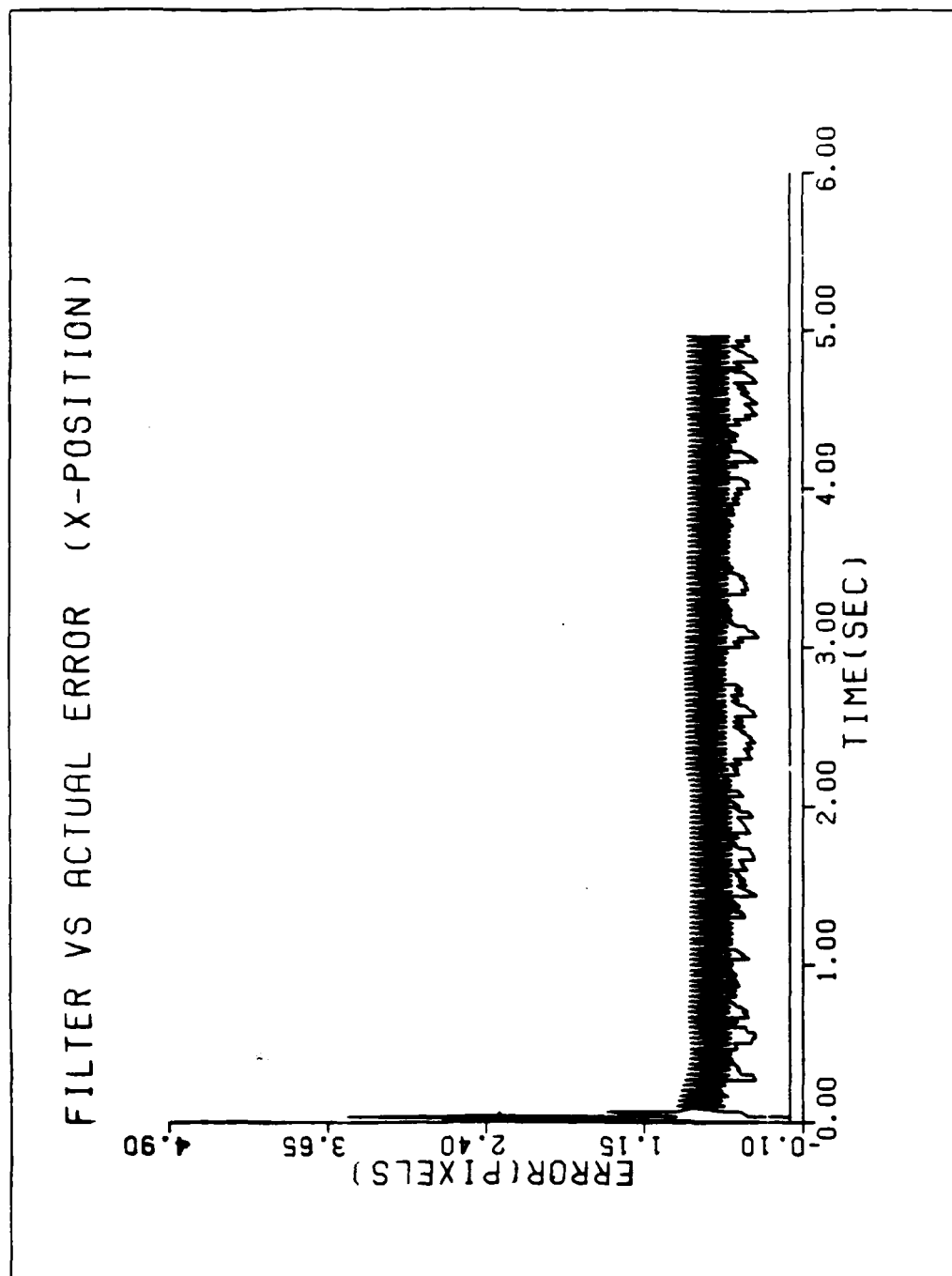


Figure C1 Performance plots for T2G10S/N10

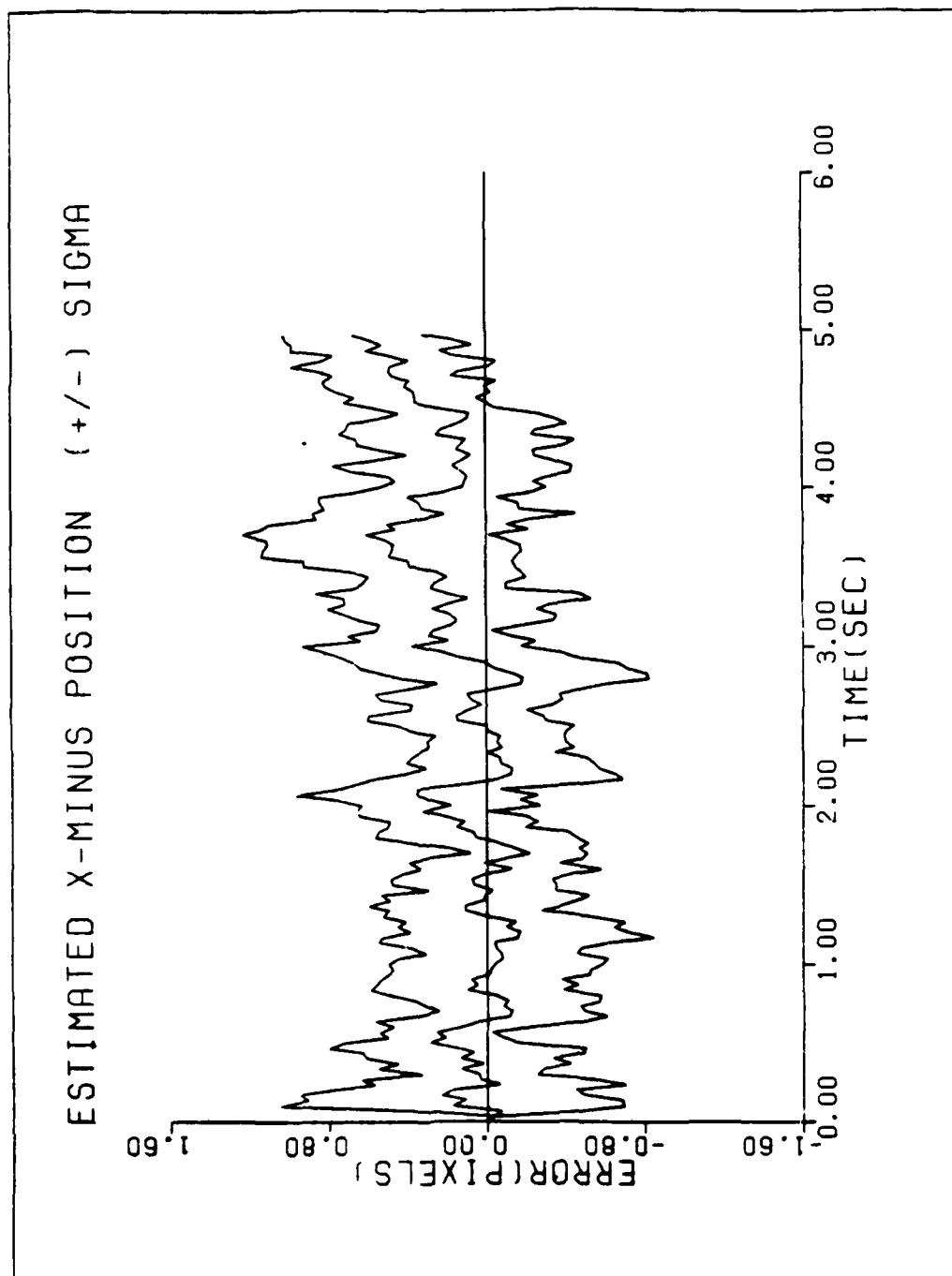


Figure C2 Performance plots for T2G10S/N10

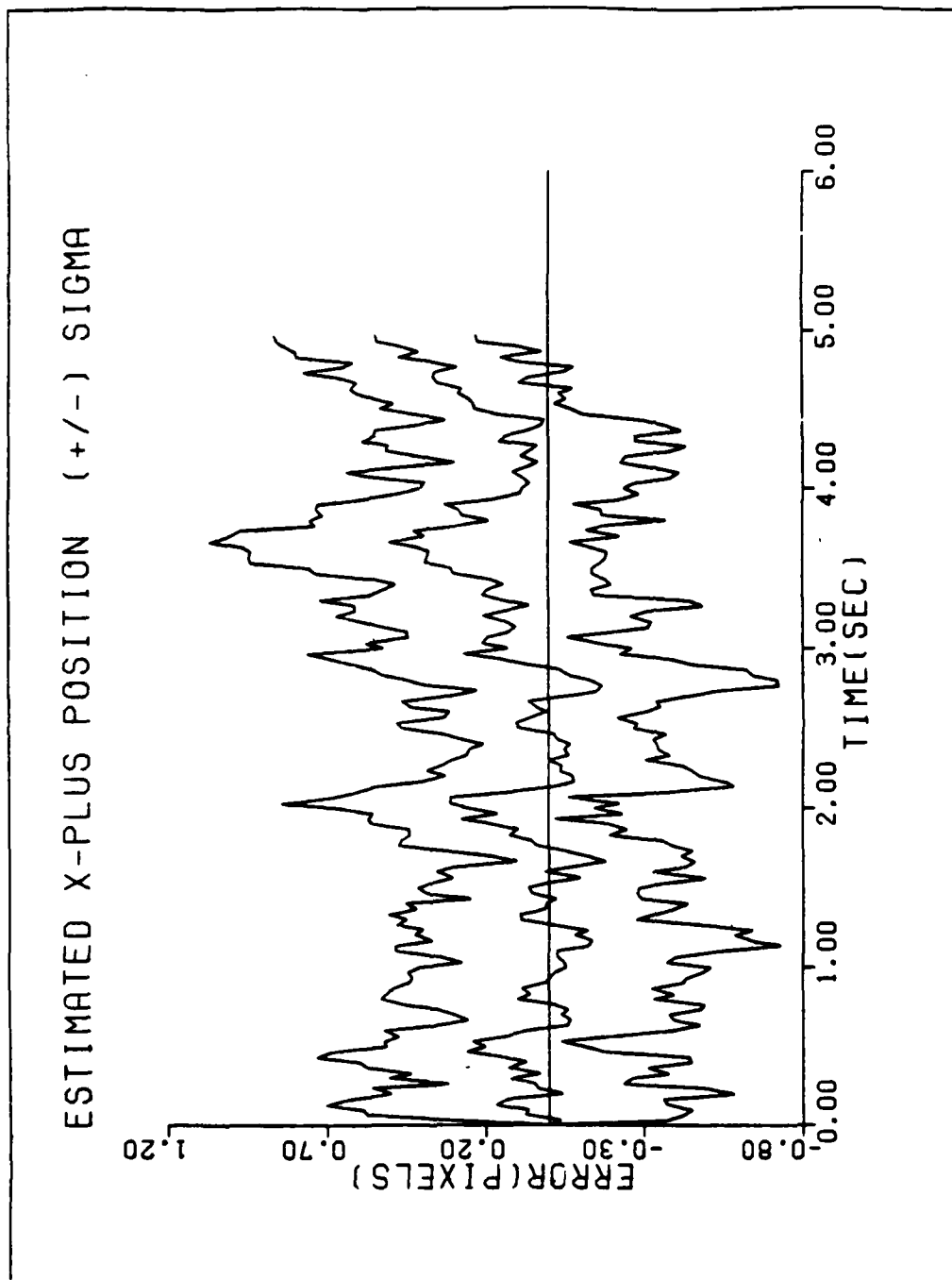


Figure C3 Performance plots for T2G10S/N10

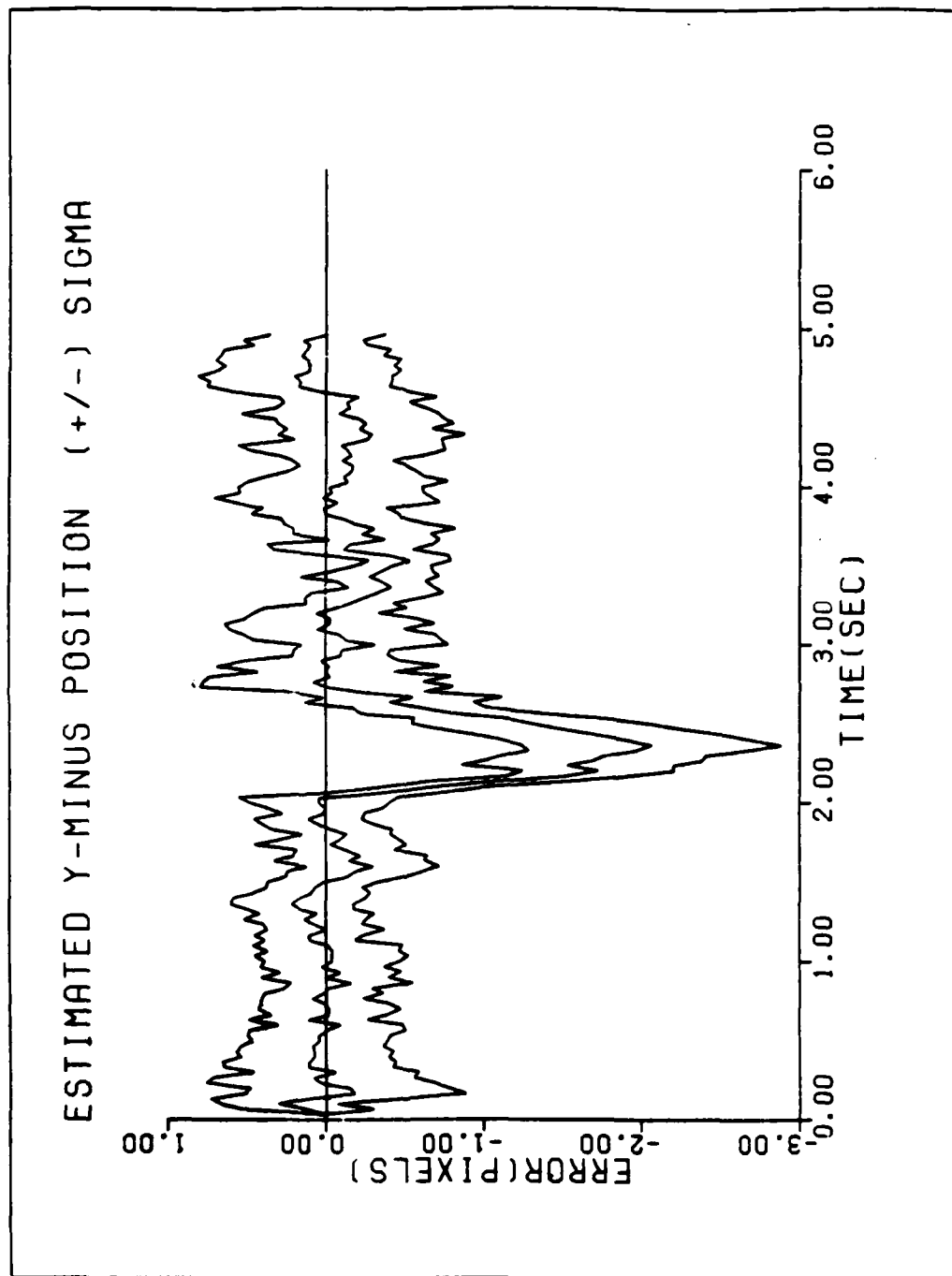


Figure C4 Performance plots for T2G10S/N10

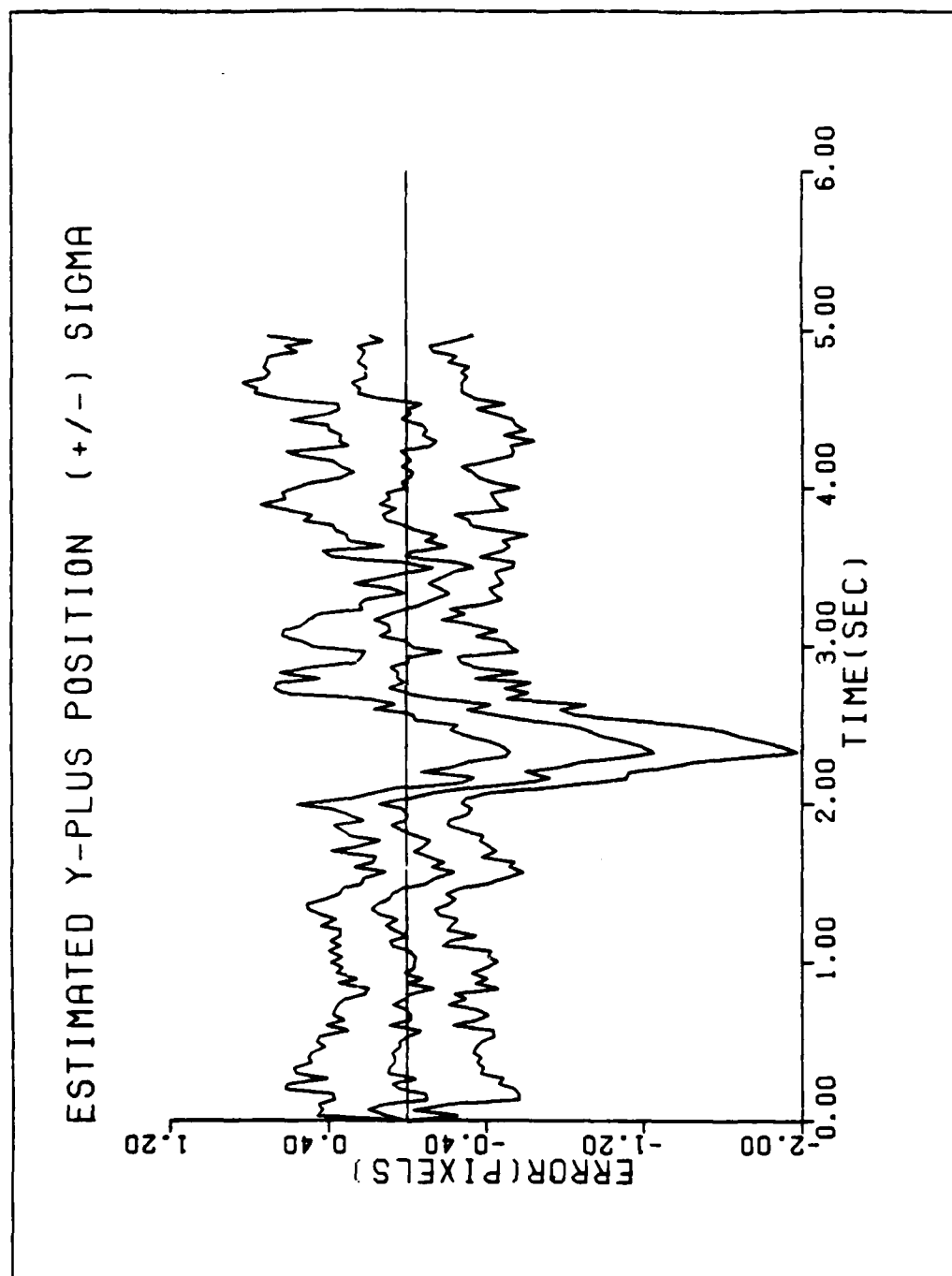


Figure C5 Performance plots for T2G10S/N10

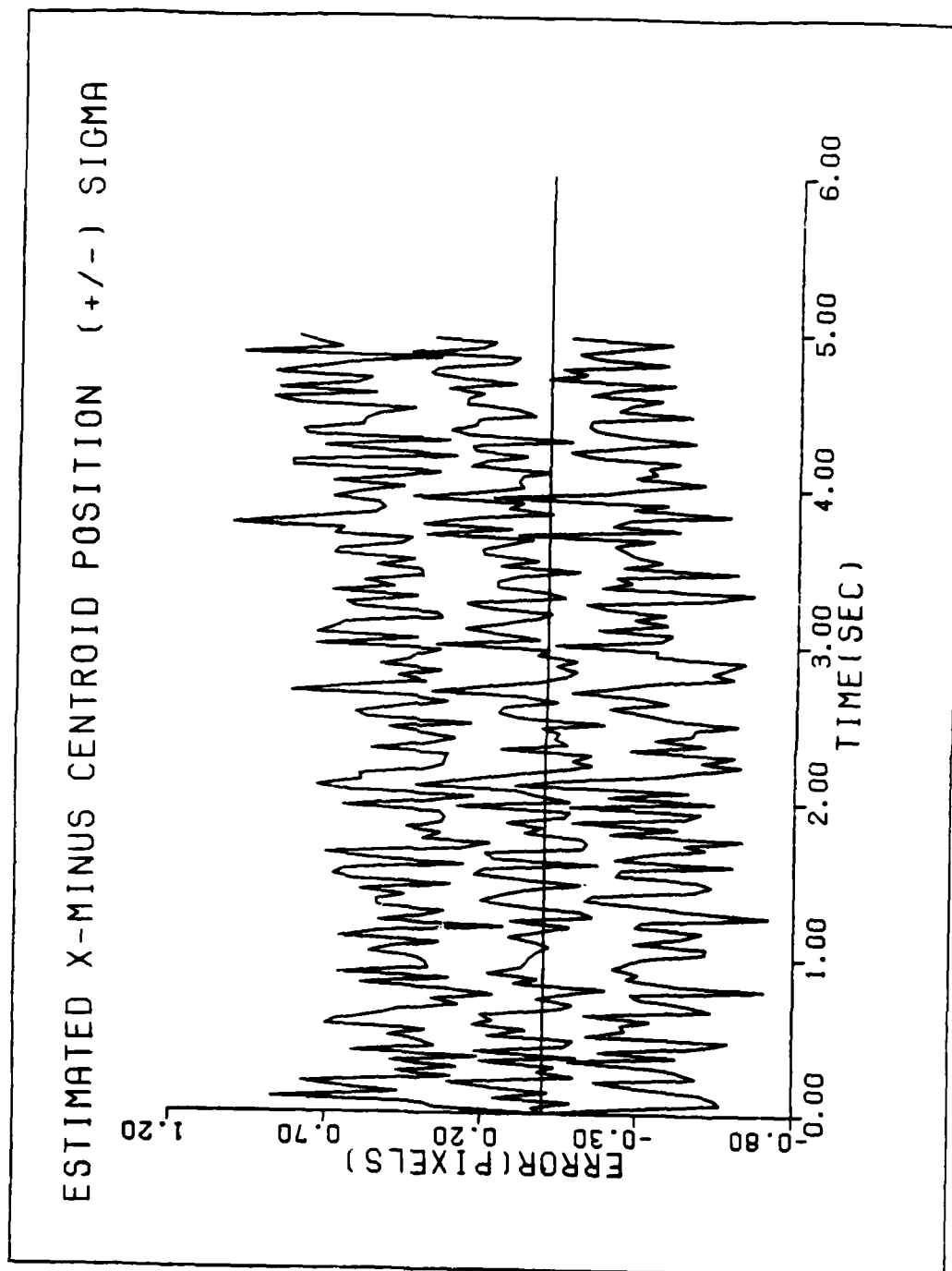


Figure C6 Performance plots for T2G10S/N10

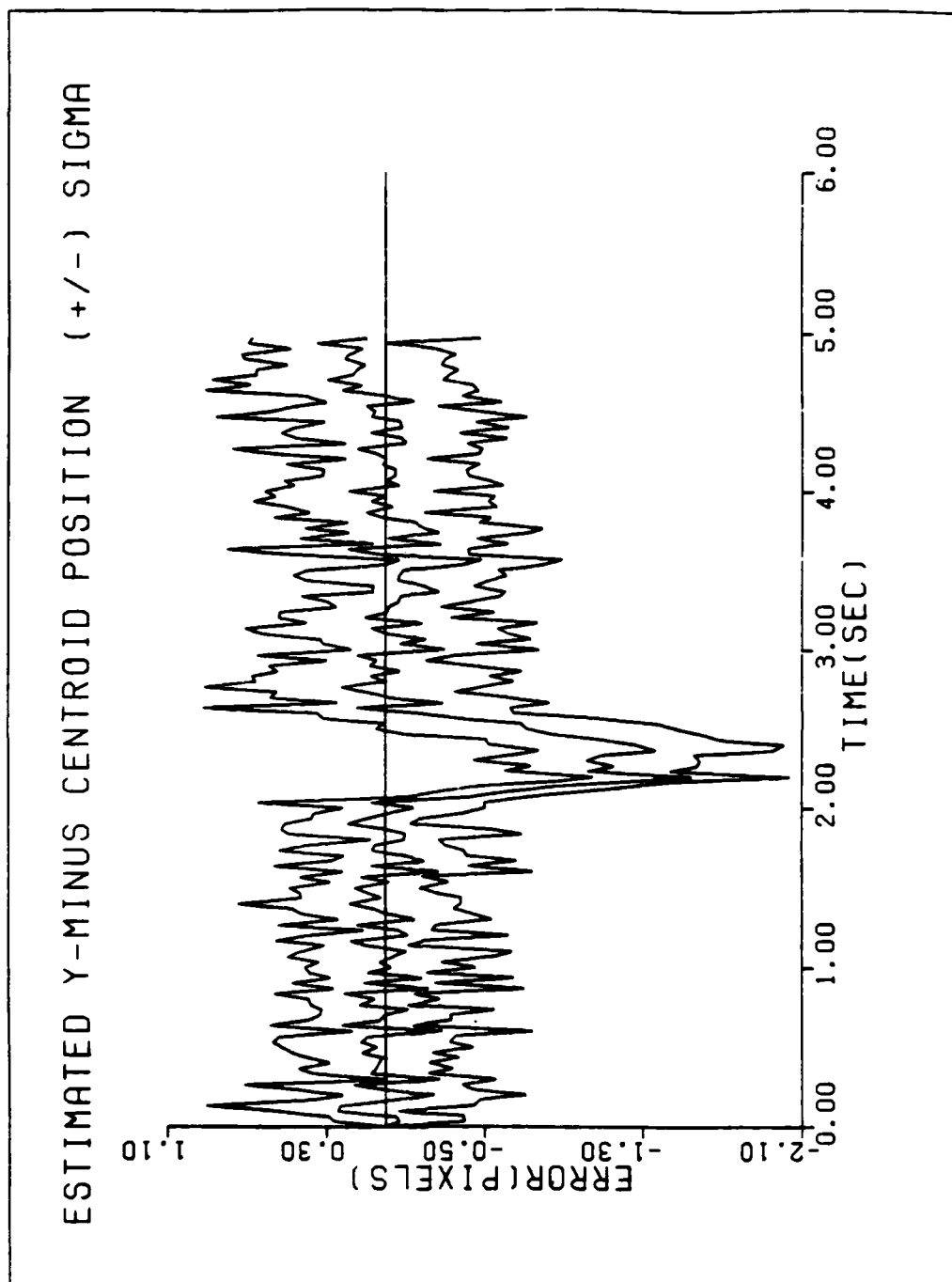


Figure C7 Performance plots for T2G10S/N10

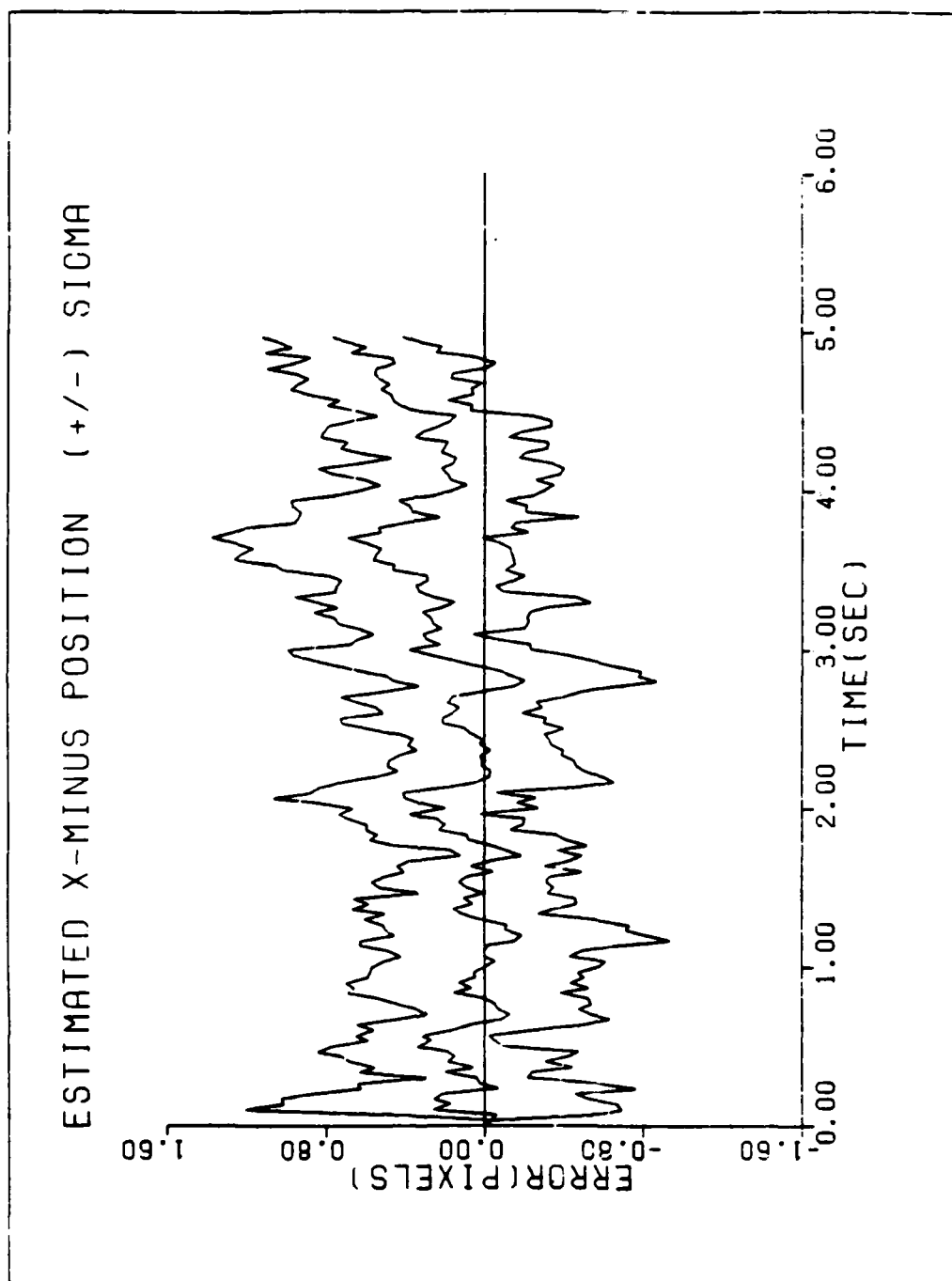


Figure C8 Performance plots for T2G10S/N5

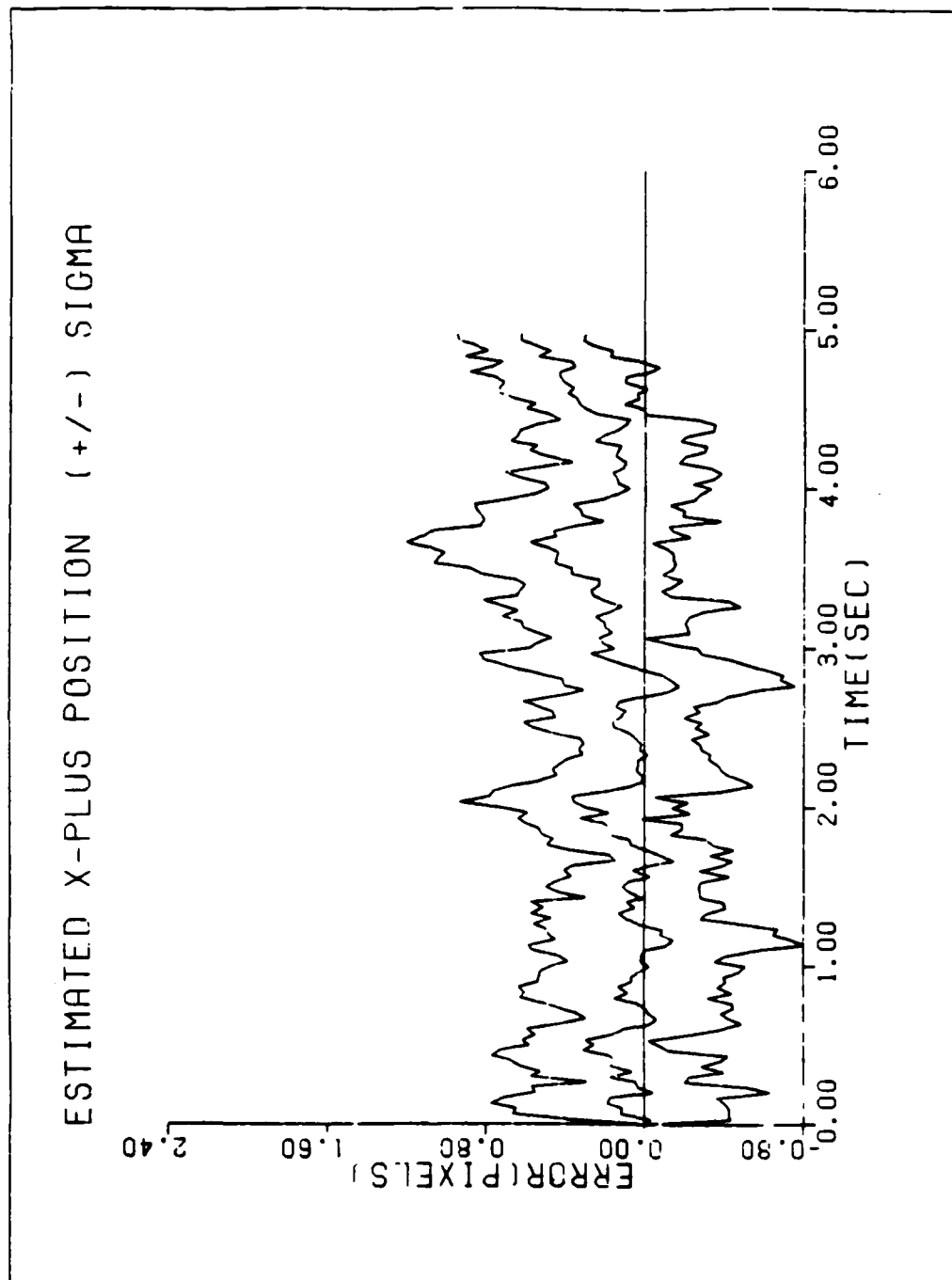


Figure C9 Performance plots for T2G10S/N5

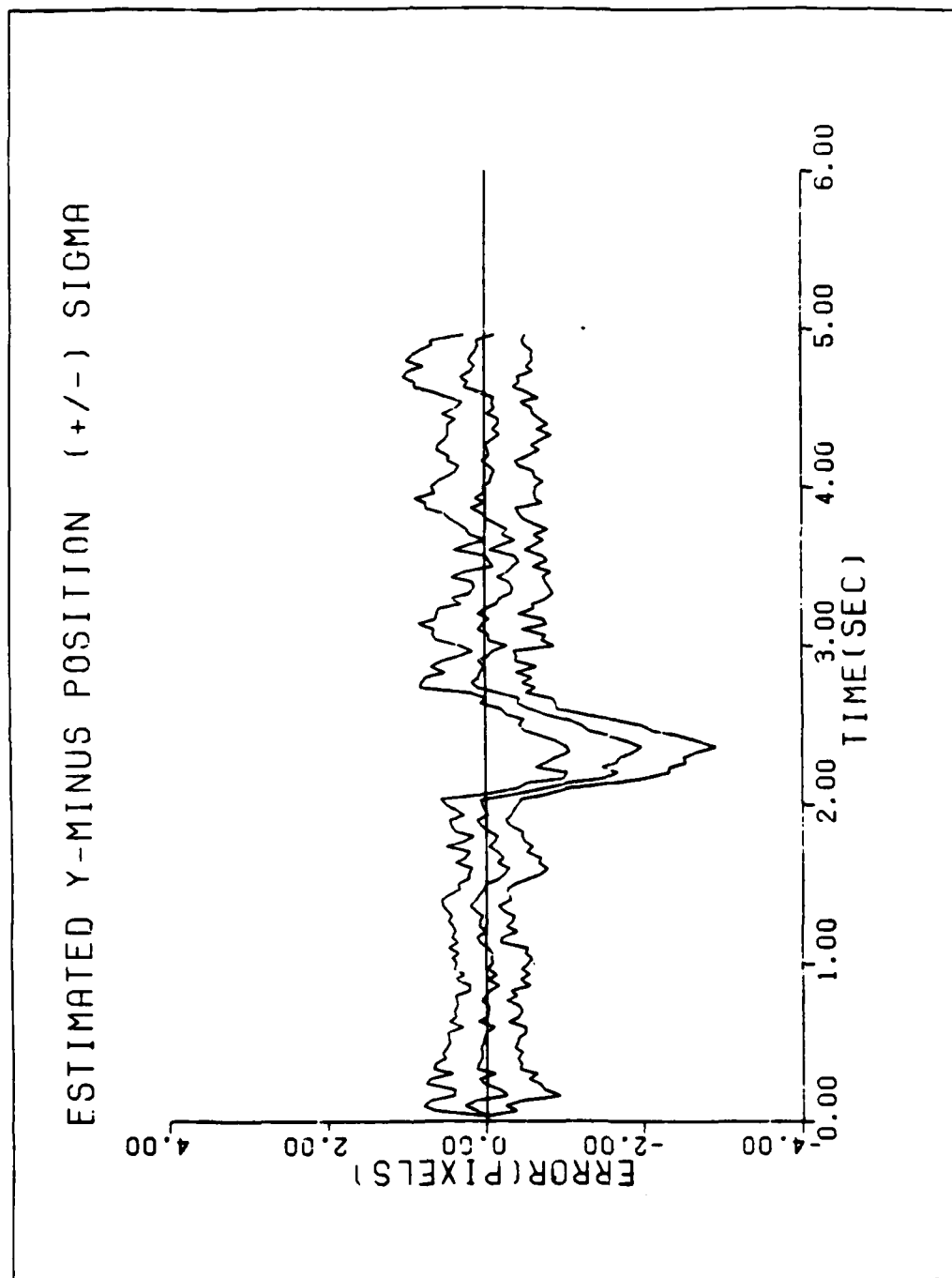


Figure C10 Performance plots for T2G10S/N5

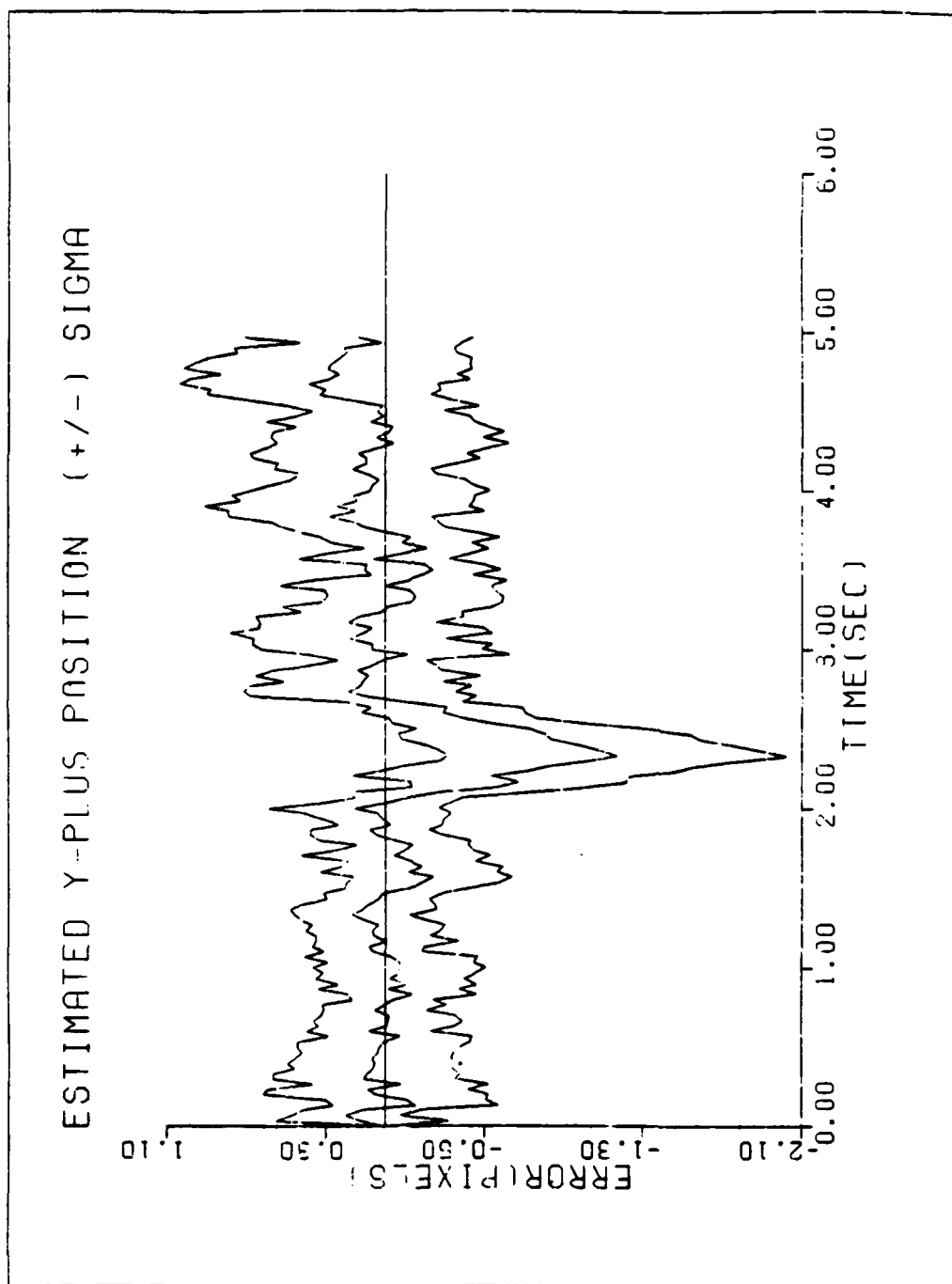


Figure C11 Performance plots for T2G10S/N5

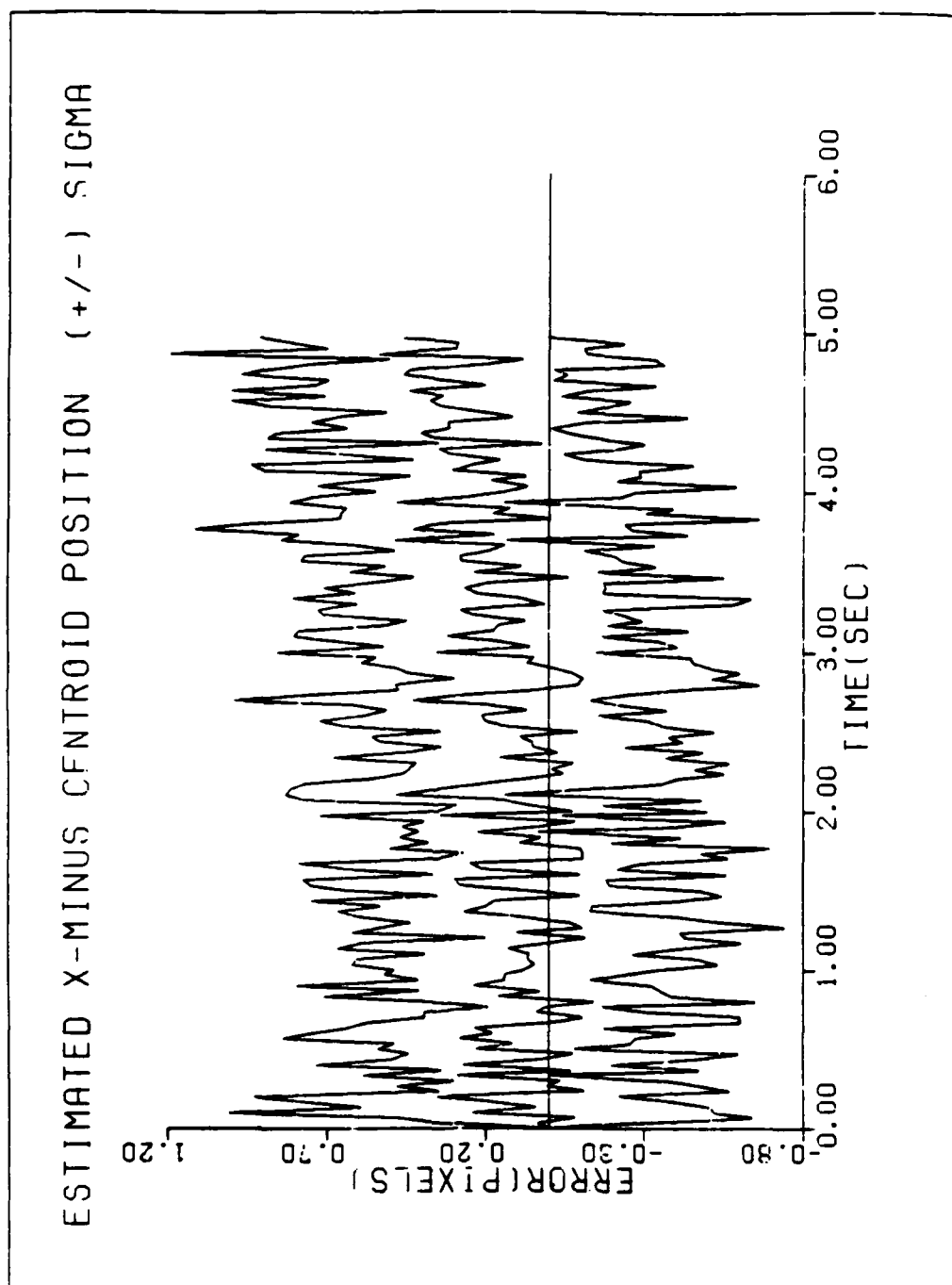


Figure C12 Performance plots for T2G10S/N5

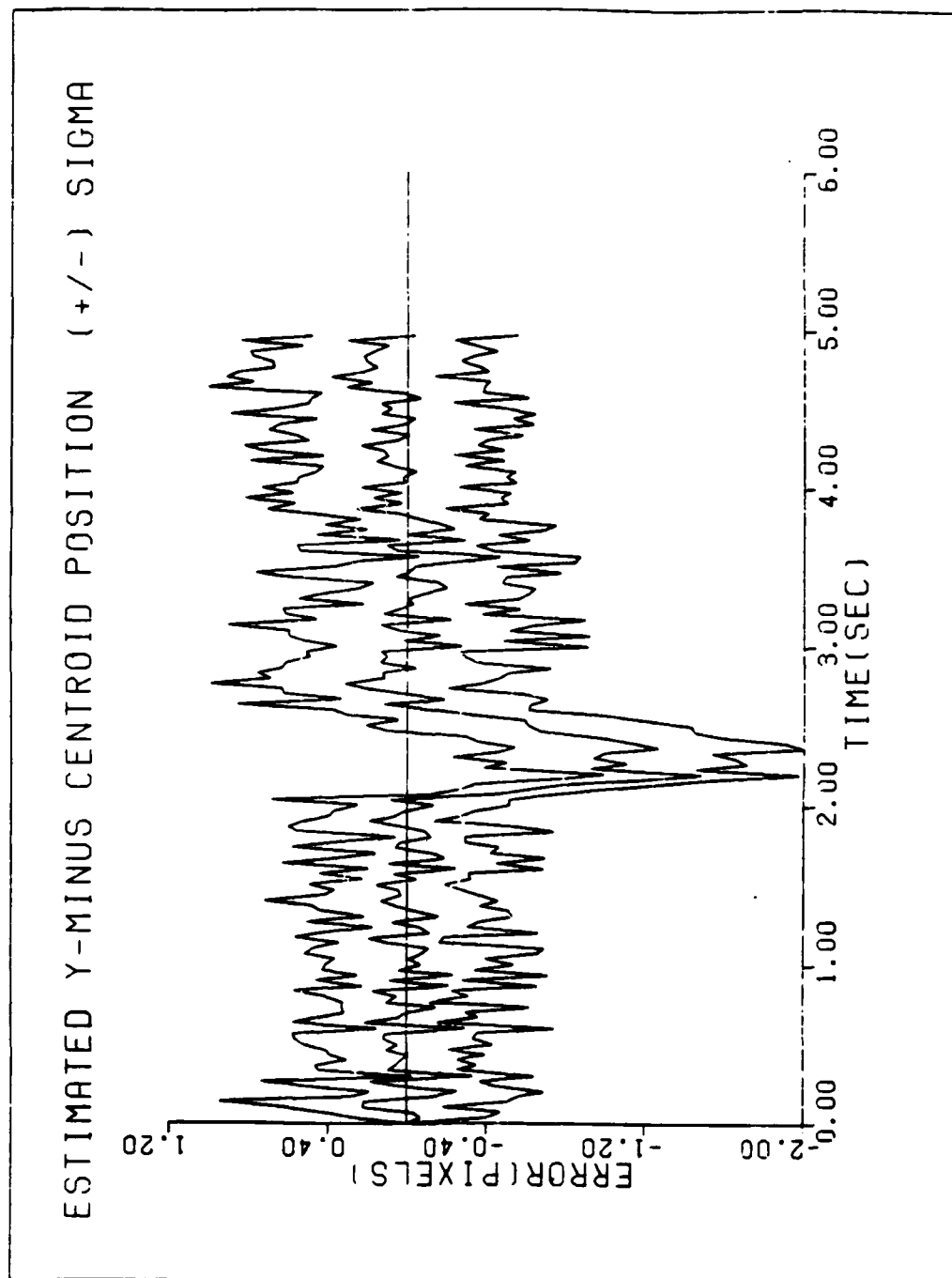


Figure C13 Performance plots for T2G10S/N5

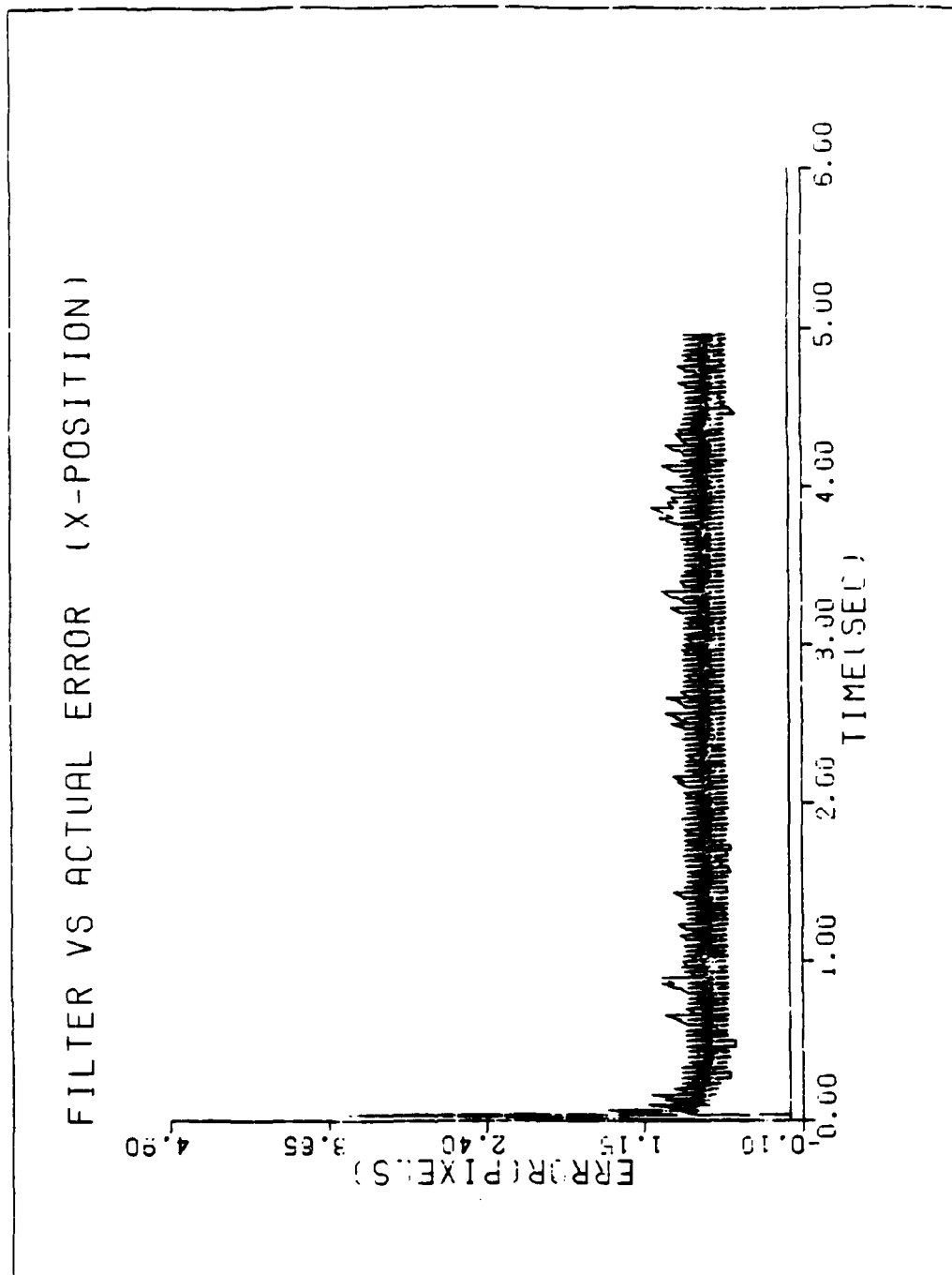


Figure C14 Performance plots for T2G10S/N2

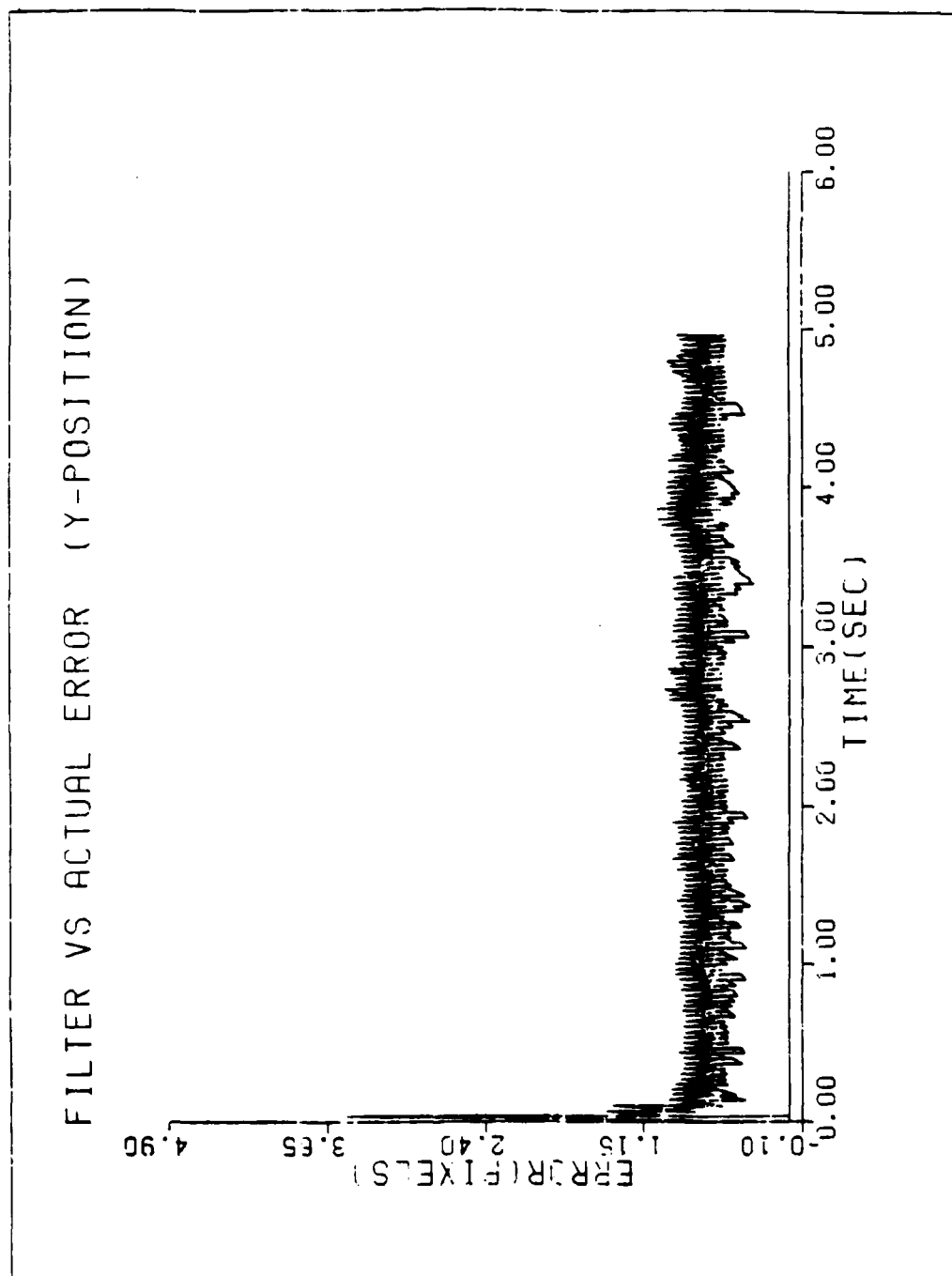


Figure C15 Performance plots for T2G10S/N2

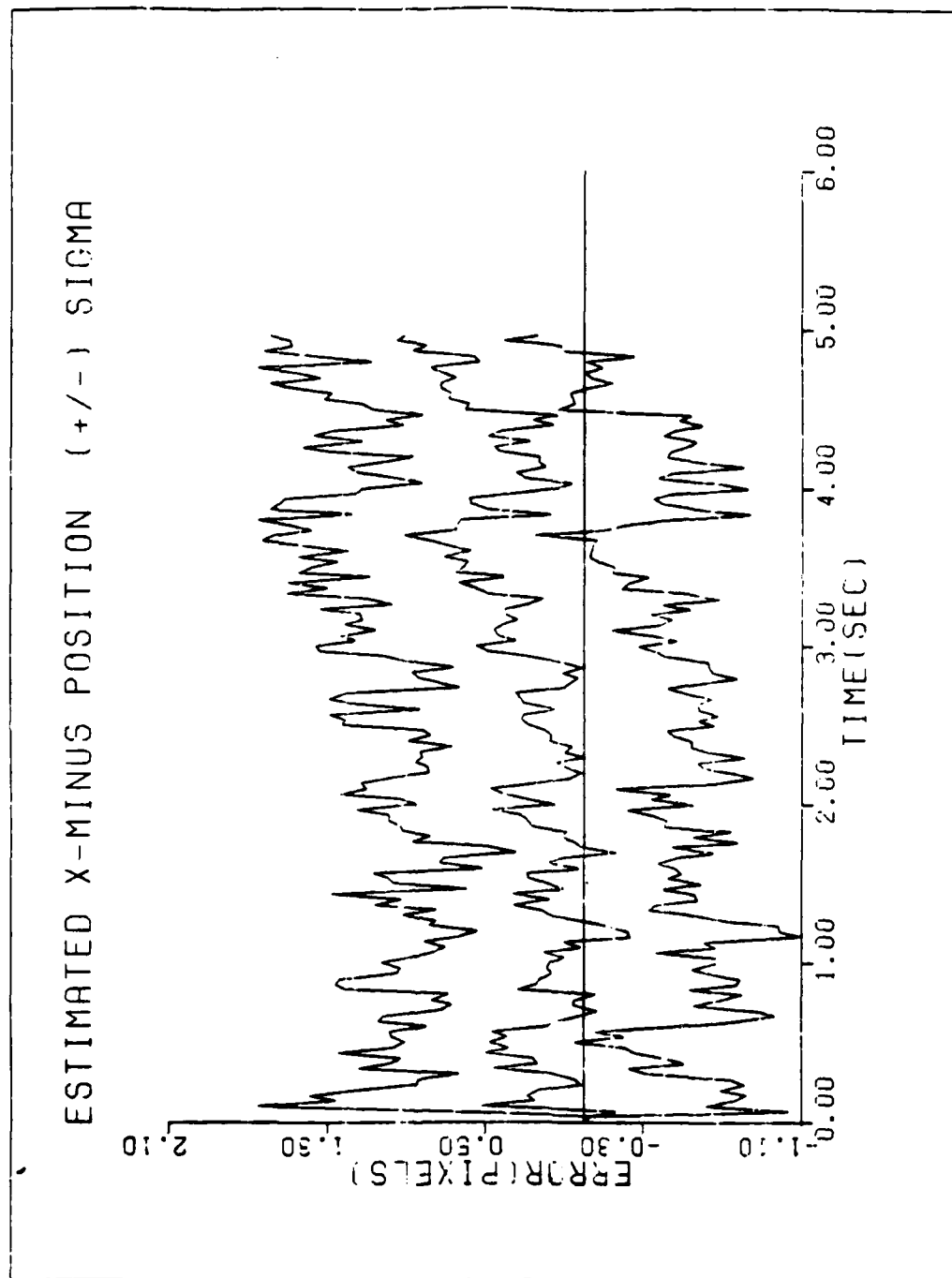


Figure C16 Performance plots for T2G10S/N2

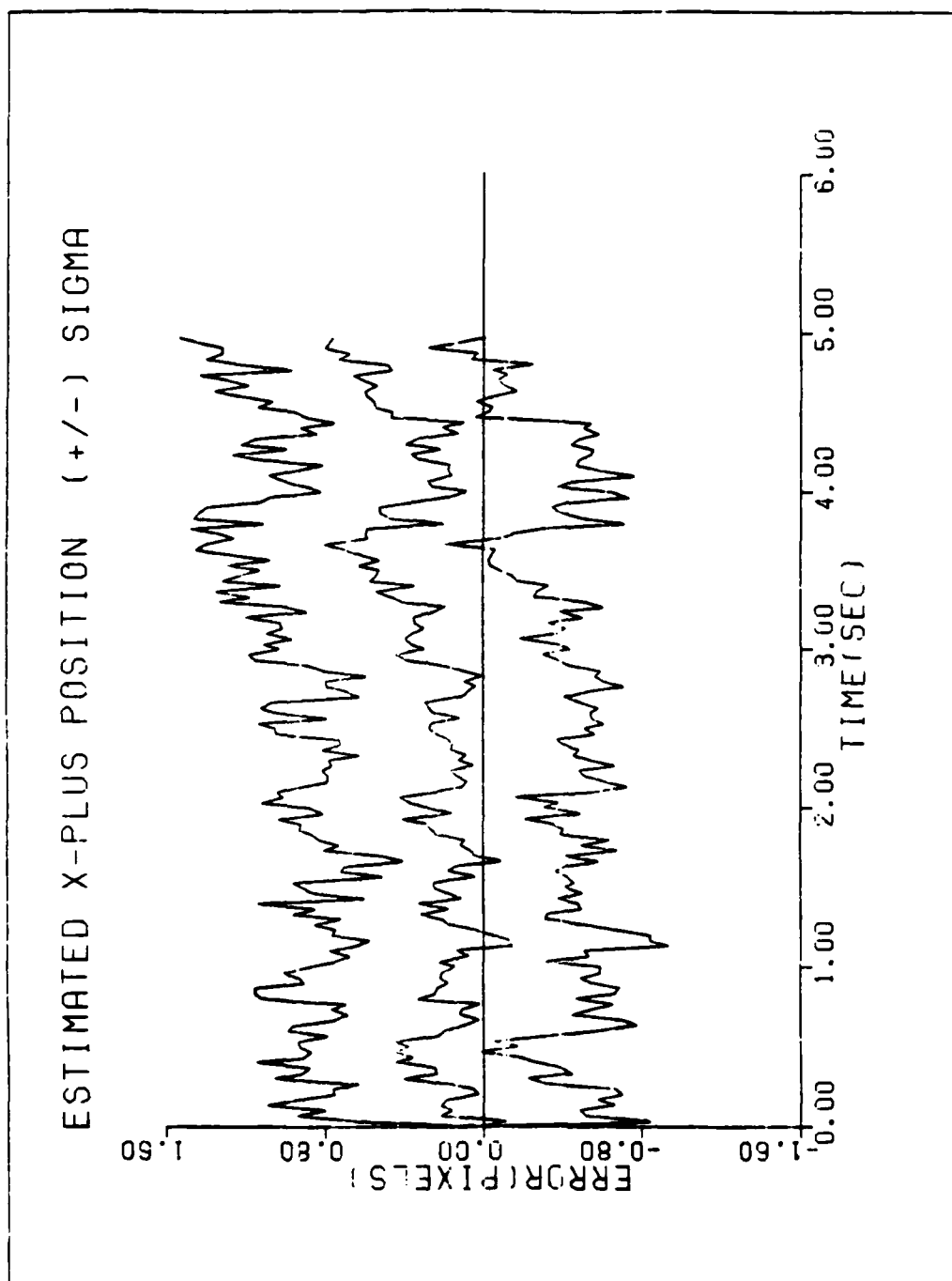


Figure C17 Performance plots for T2G10S/N2

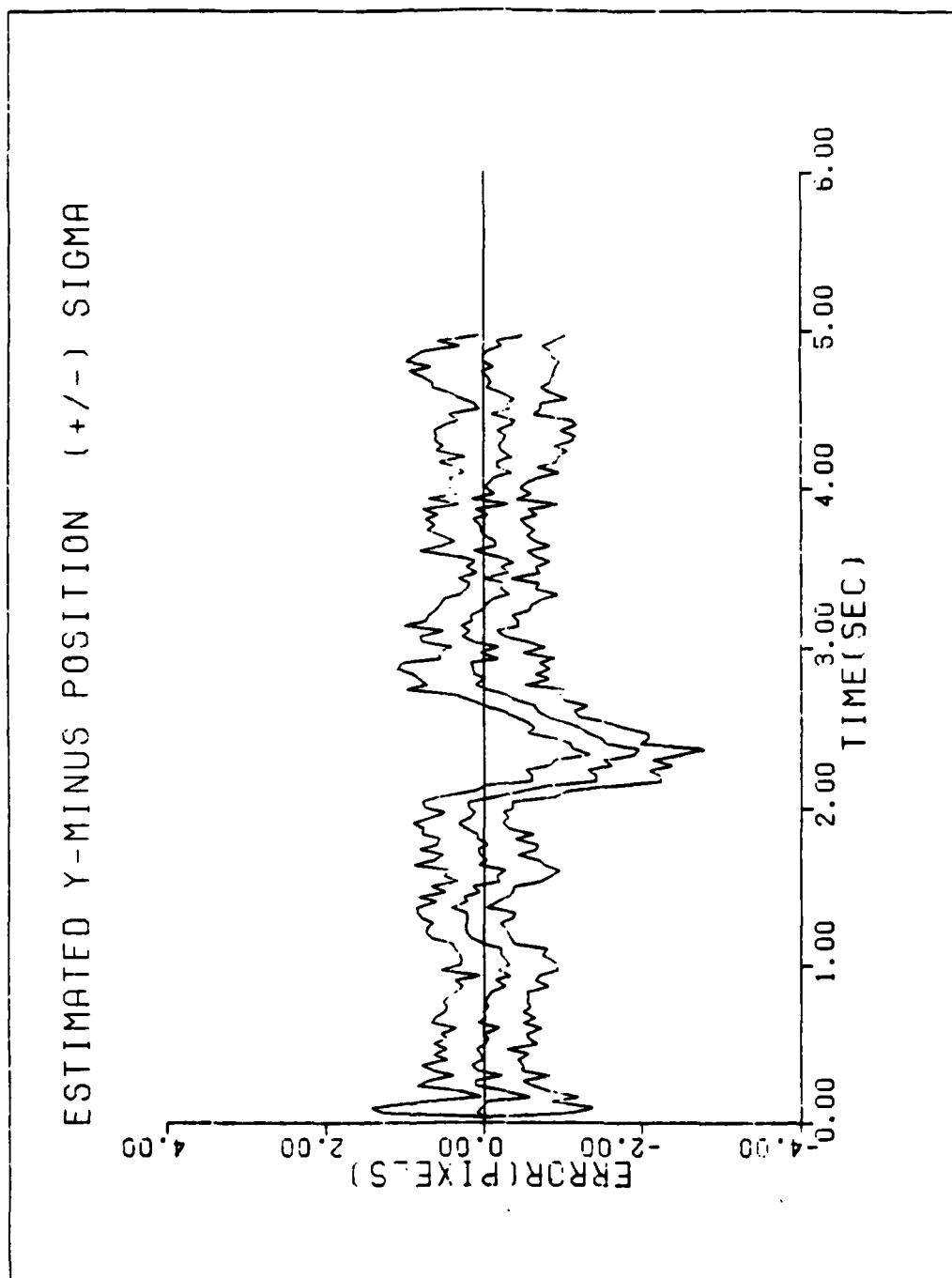


Figure C18 Performance plots for T2G10S/N2

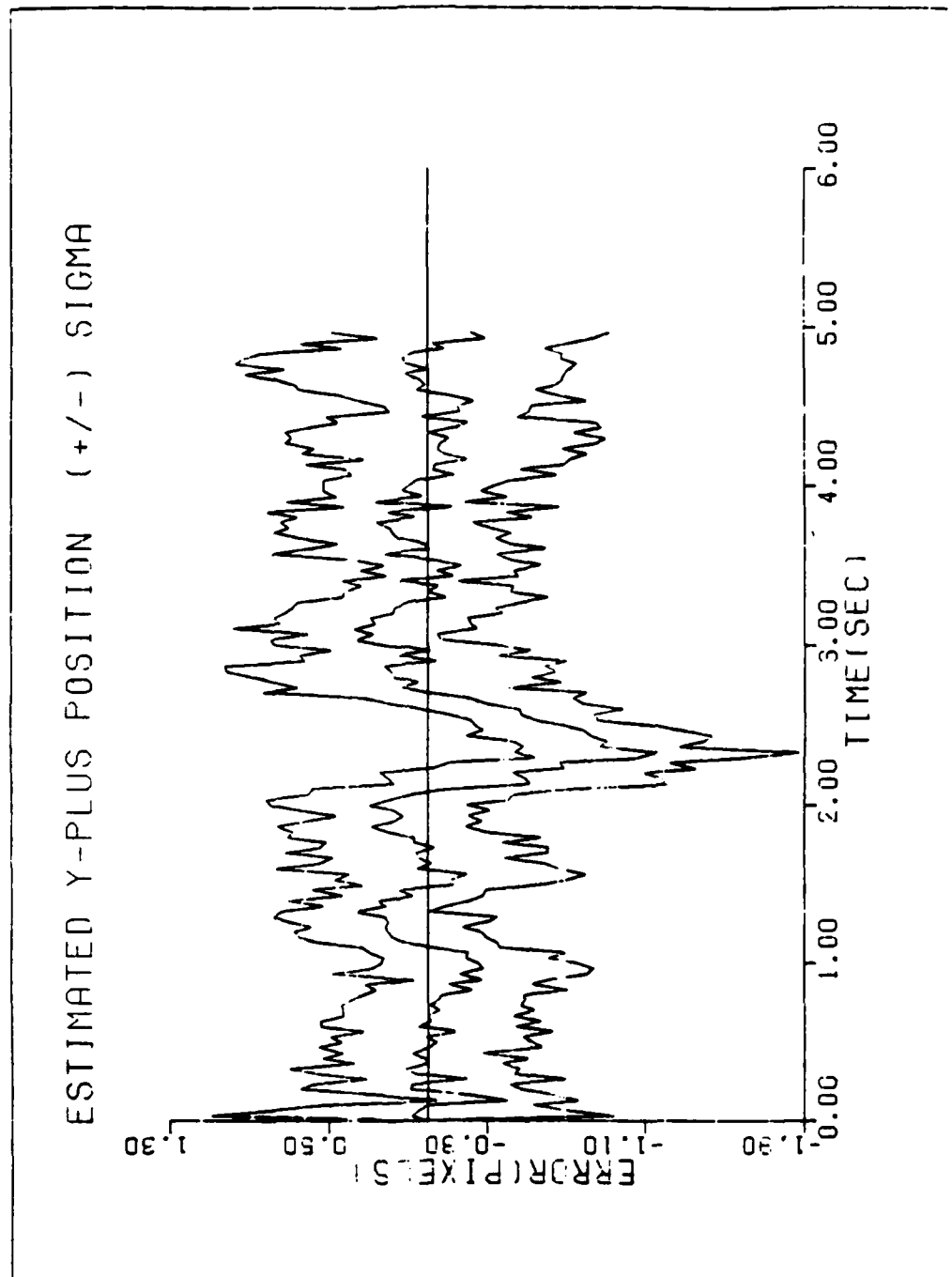


Figure C19 Performance plots for T2G10S/N2

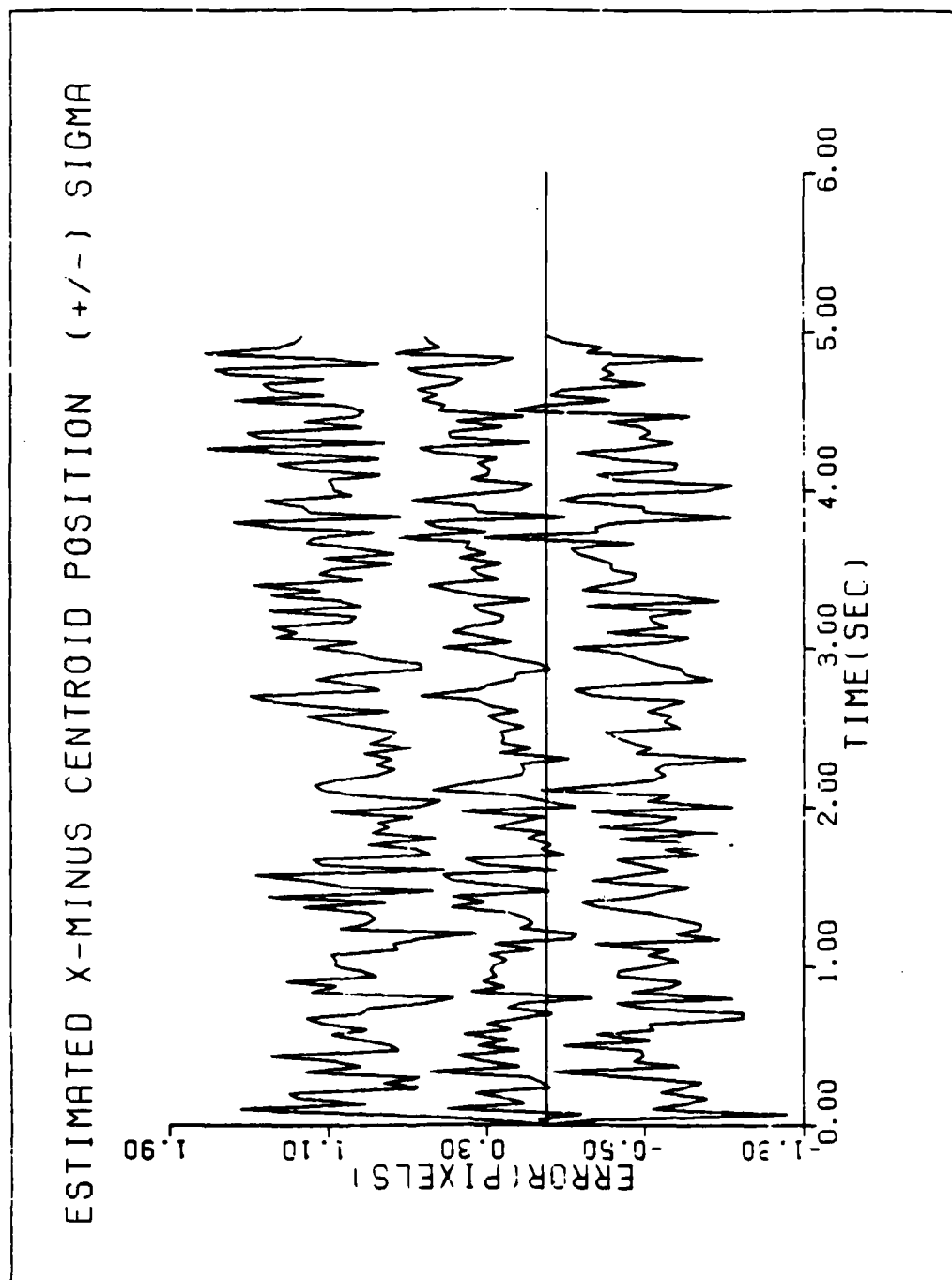


Figure C20 Performance plots for T2G10S/N2

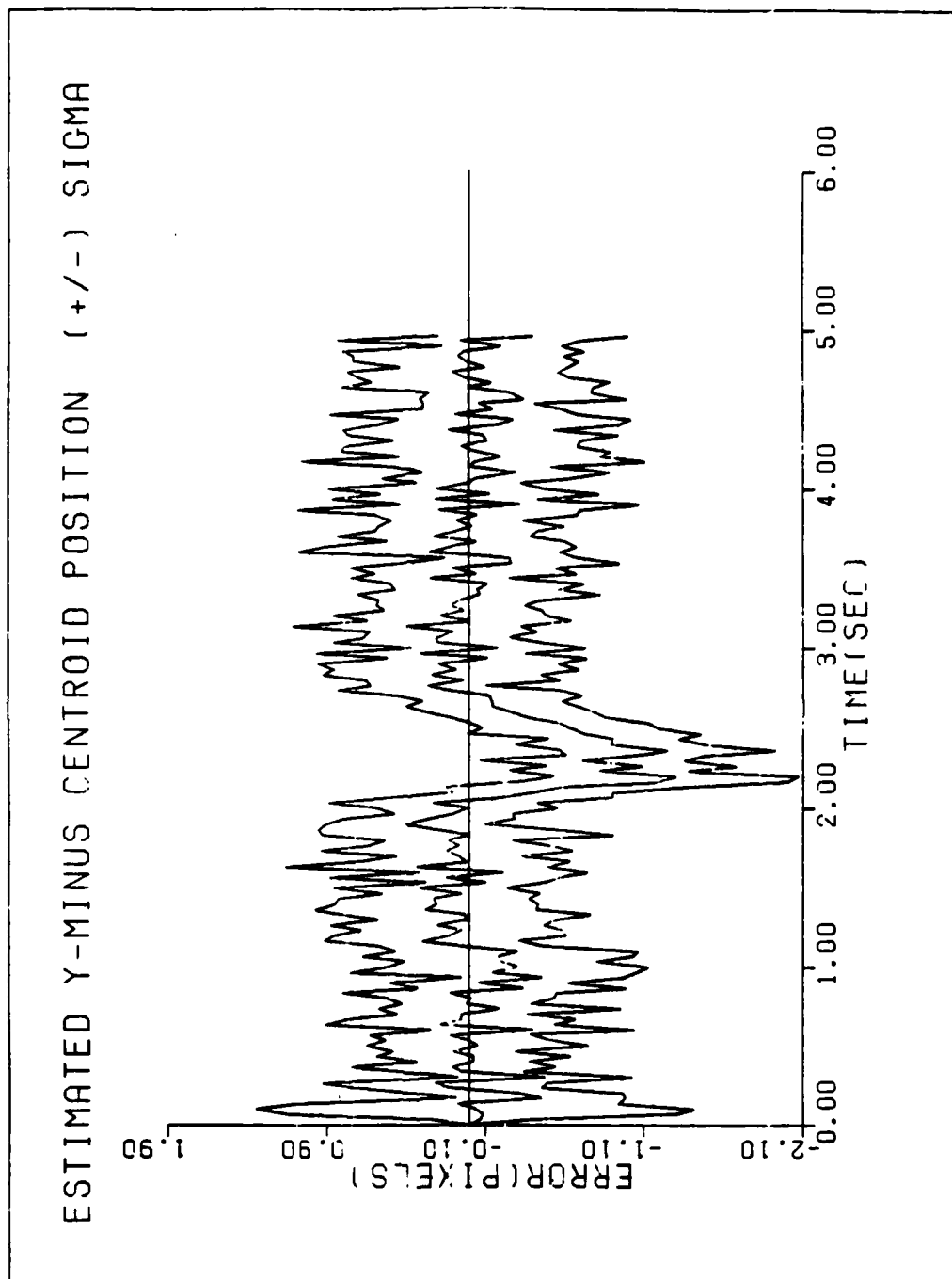


Figure C21 Performance plots for T2G10S/N2

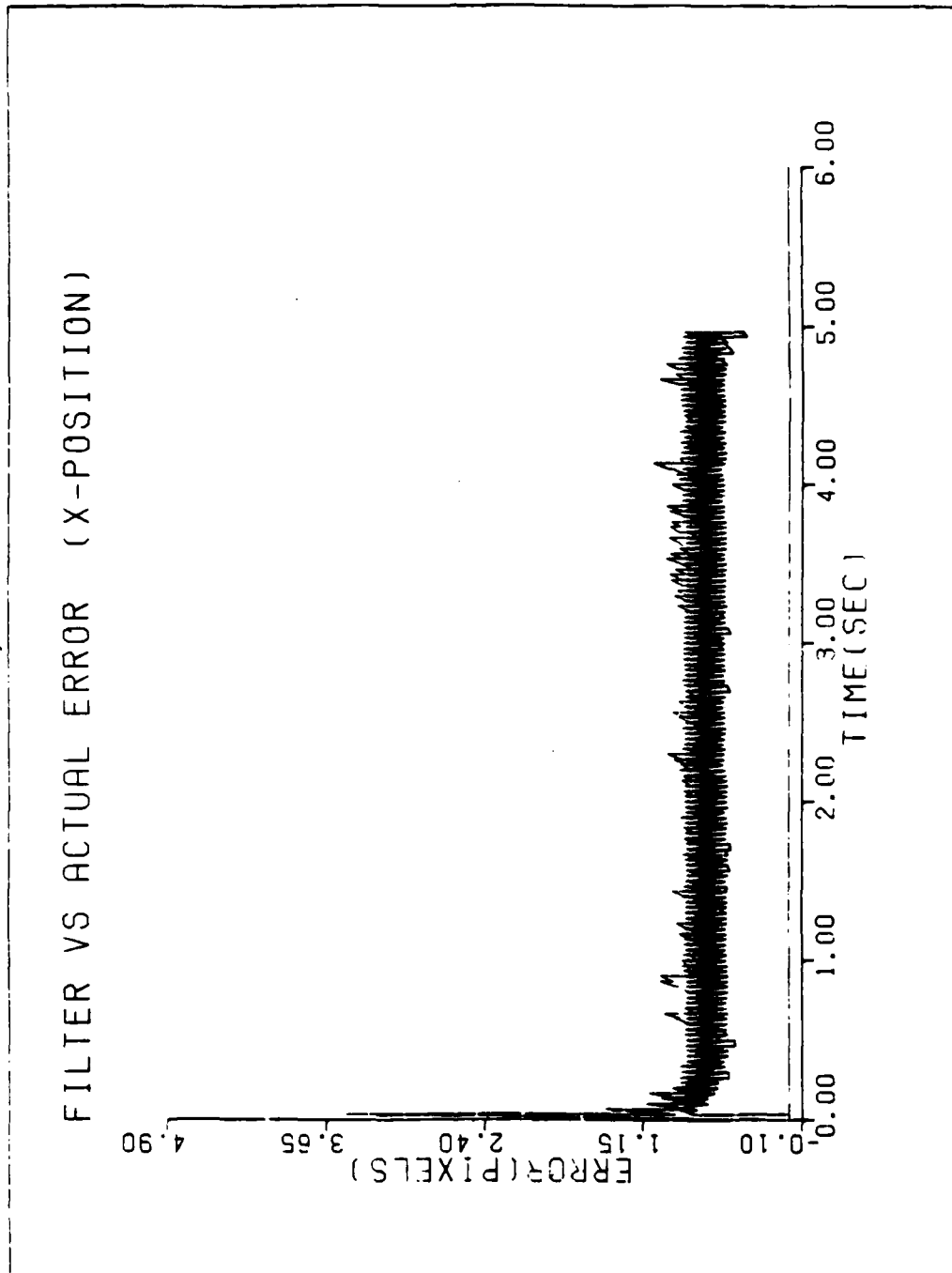


Figure C22 Performance plots for T2G20S/N2

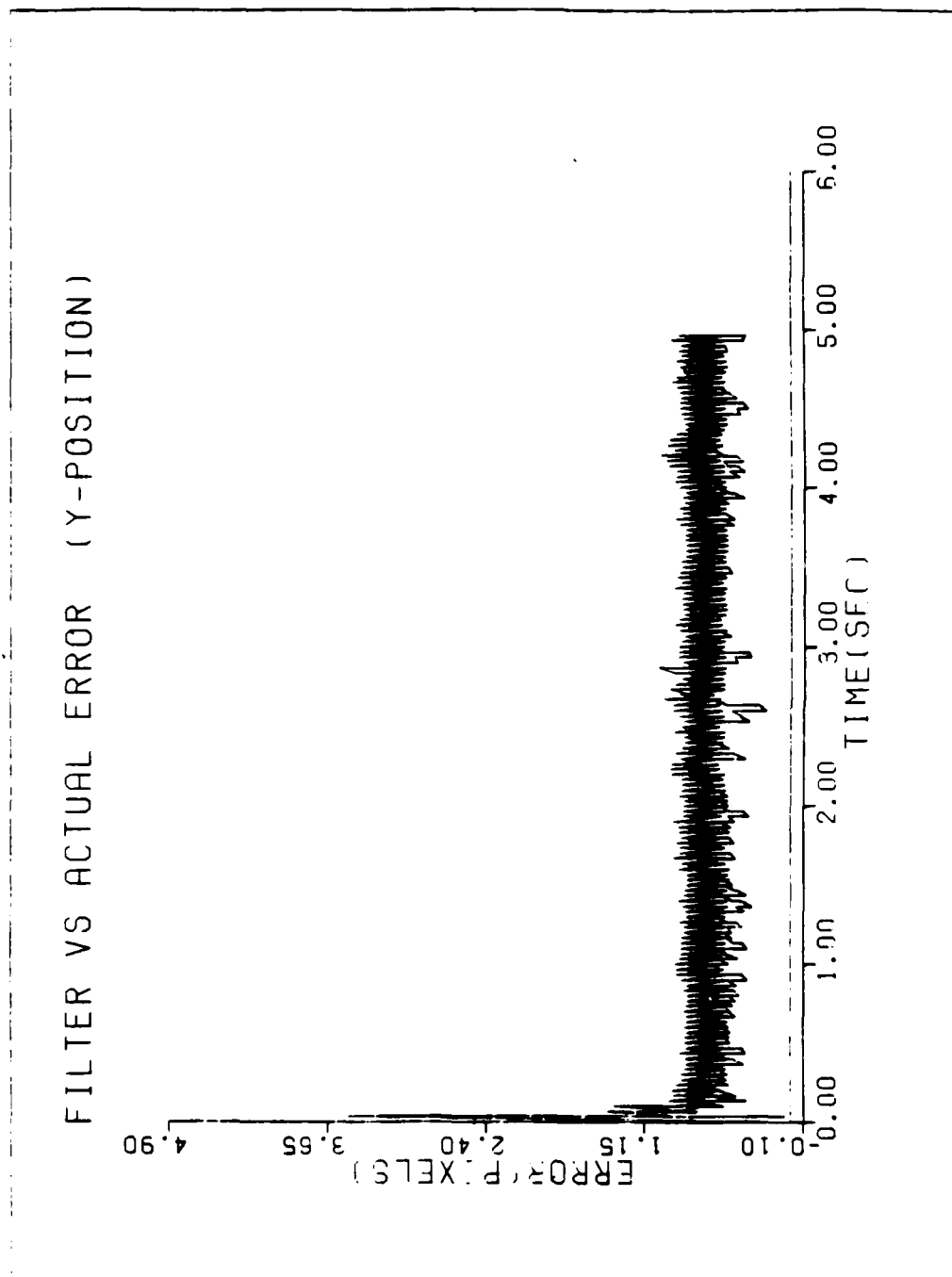


Figure C23 Performance plots for T2G20S/N2

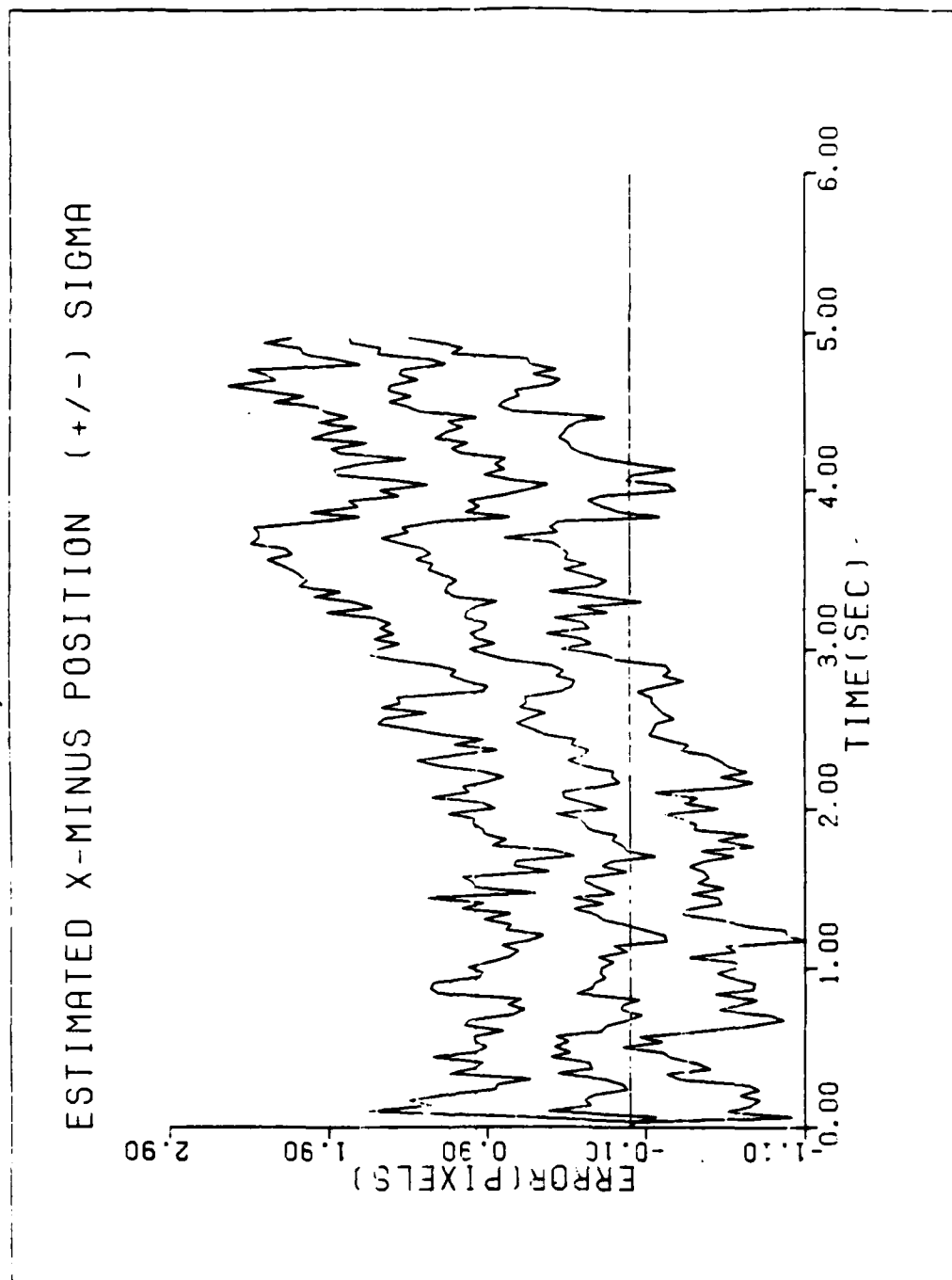


Figure C24 Performance plots for T2G20S/N2

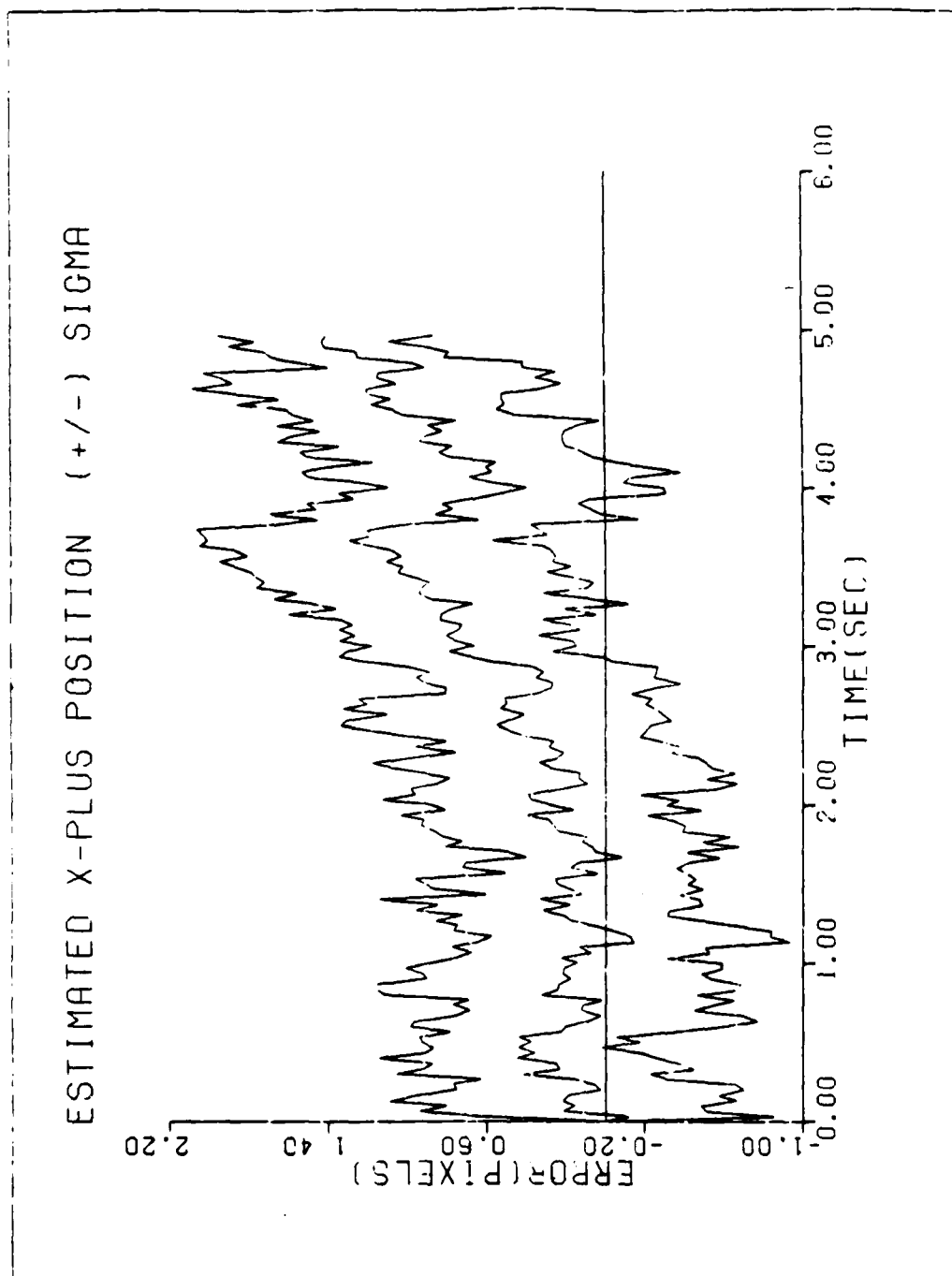


Figure C25 Performance plots for T2G20S/N2

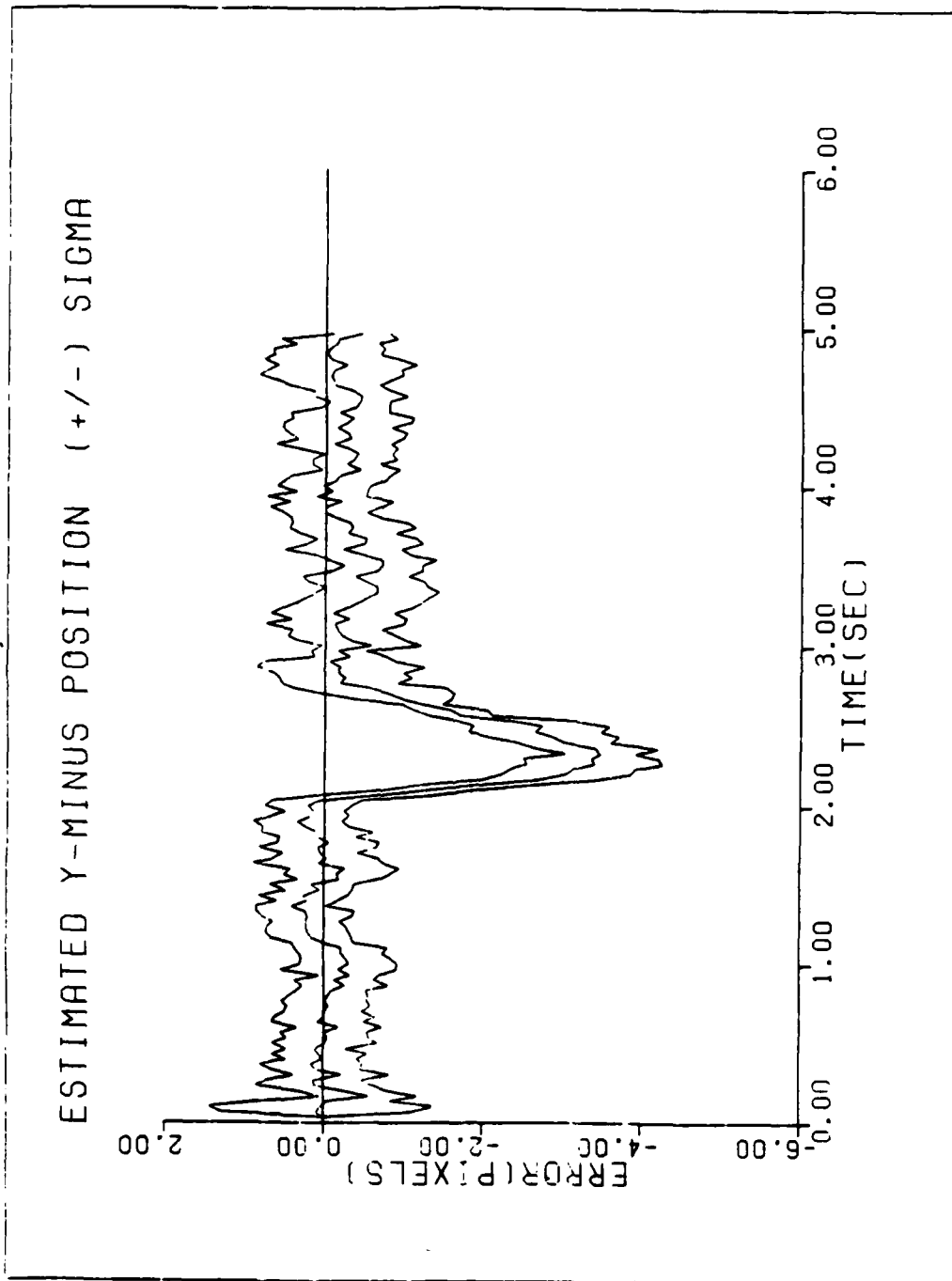


Figure C26 Performance plots for T2G20S/N2

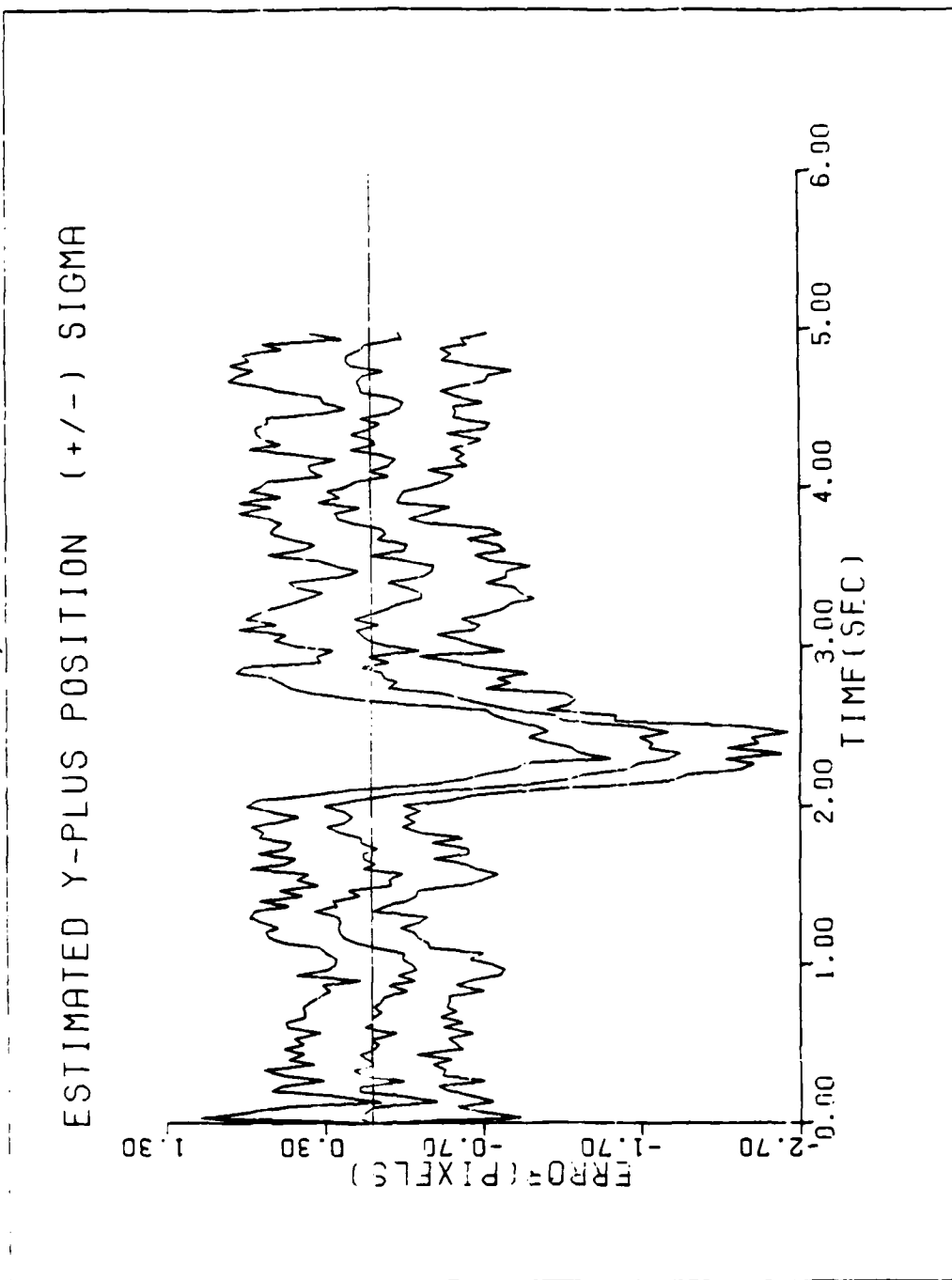


Figure C27 Performance plots for T2G20S/N2

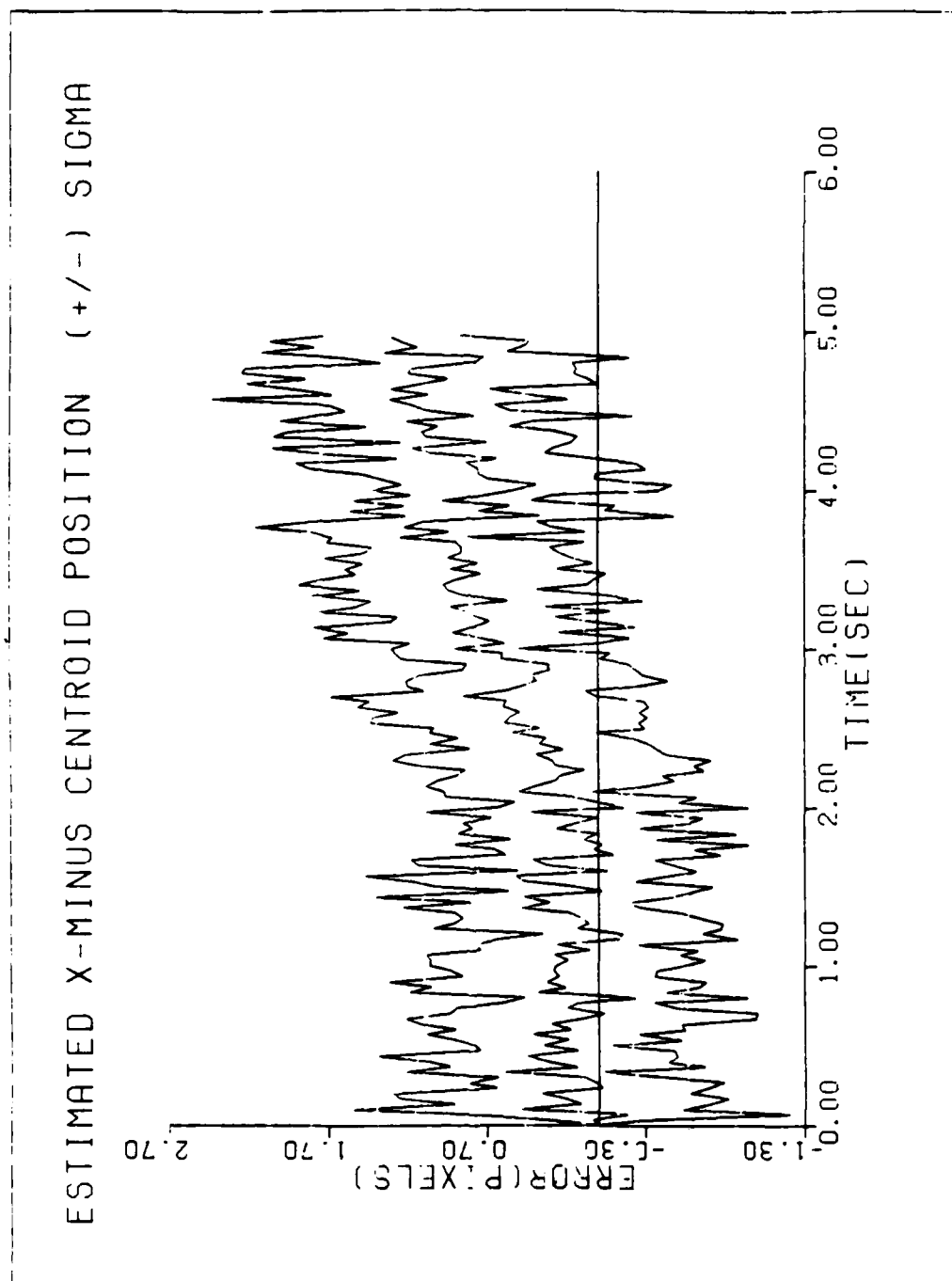


Figure C28 Performance plots for T2G20S/N2

ESTIMATED Y-MINUS CENTROID POSITION (+/-) SIGMA

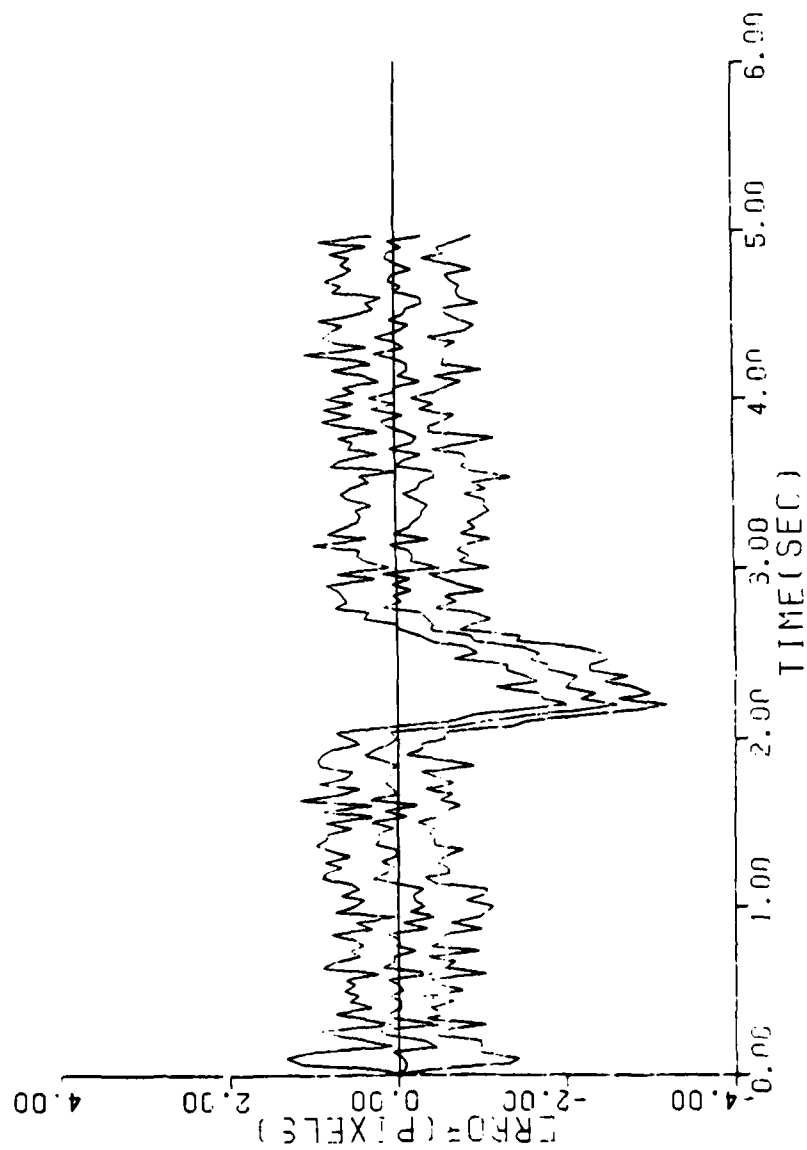


Figure C29 Performance plots for T2G20S/N2

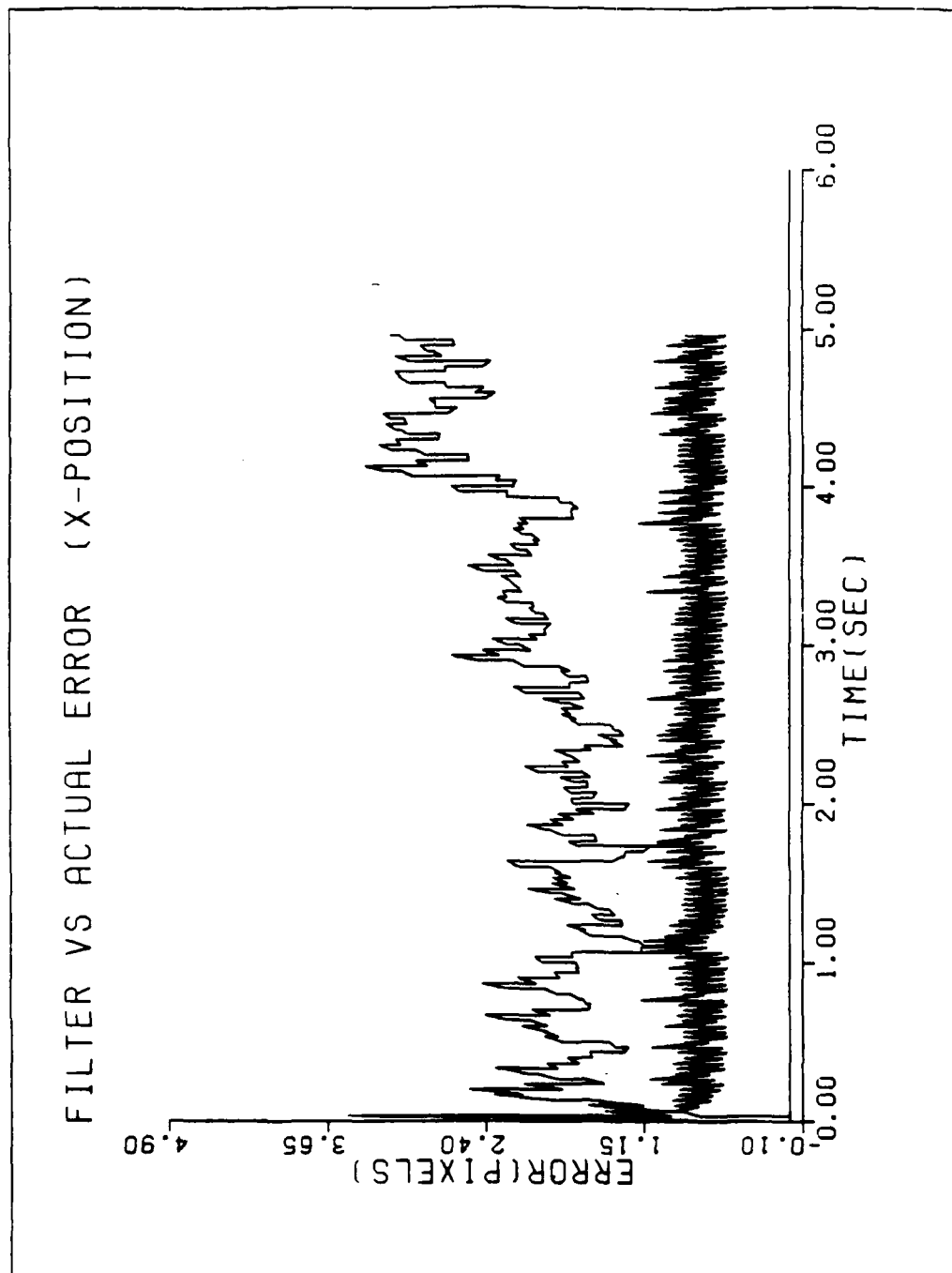


Figure C30 Performance plots for T2G10S/N1.25

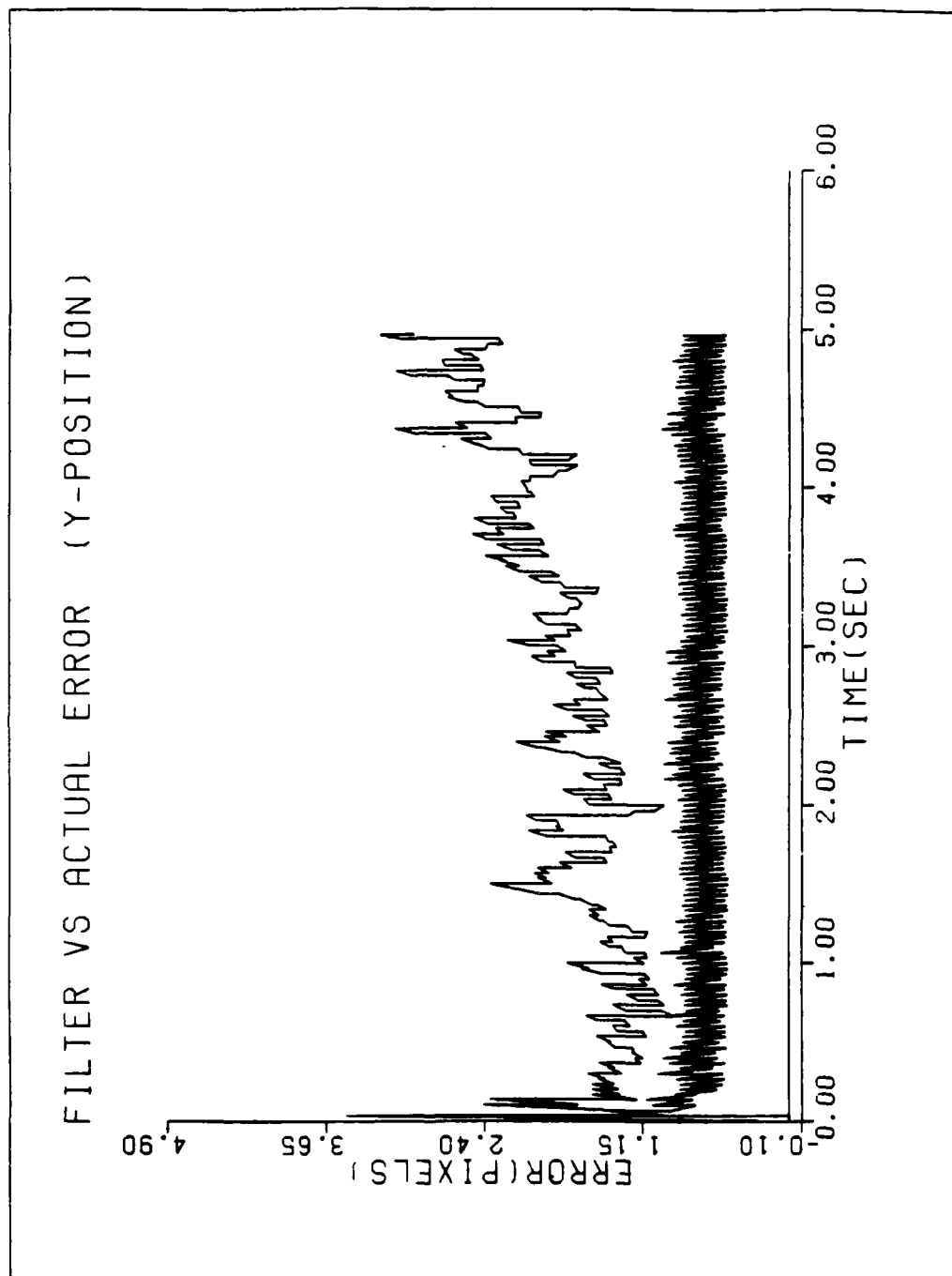


Figure C31 Performance plots for T2G10S/N1.25

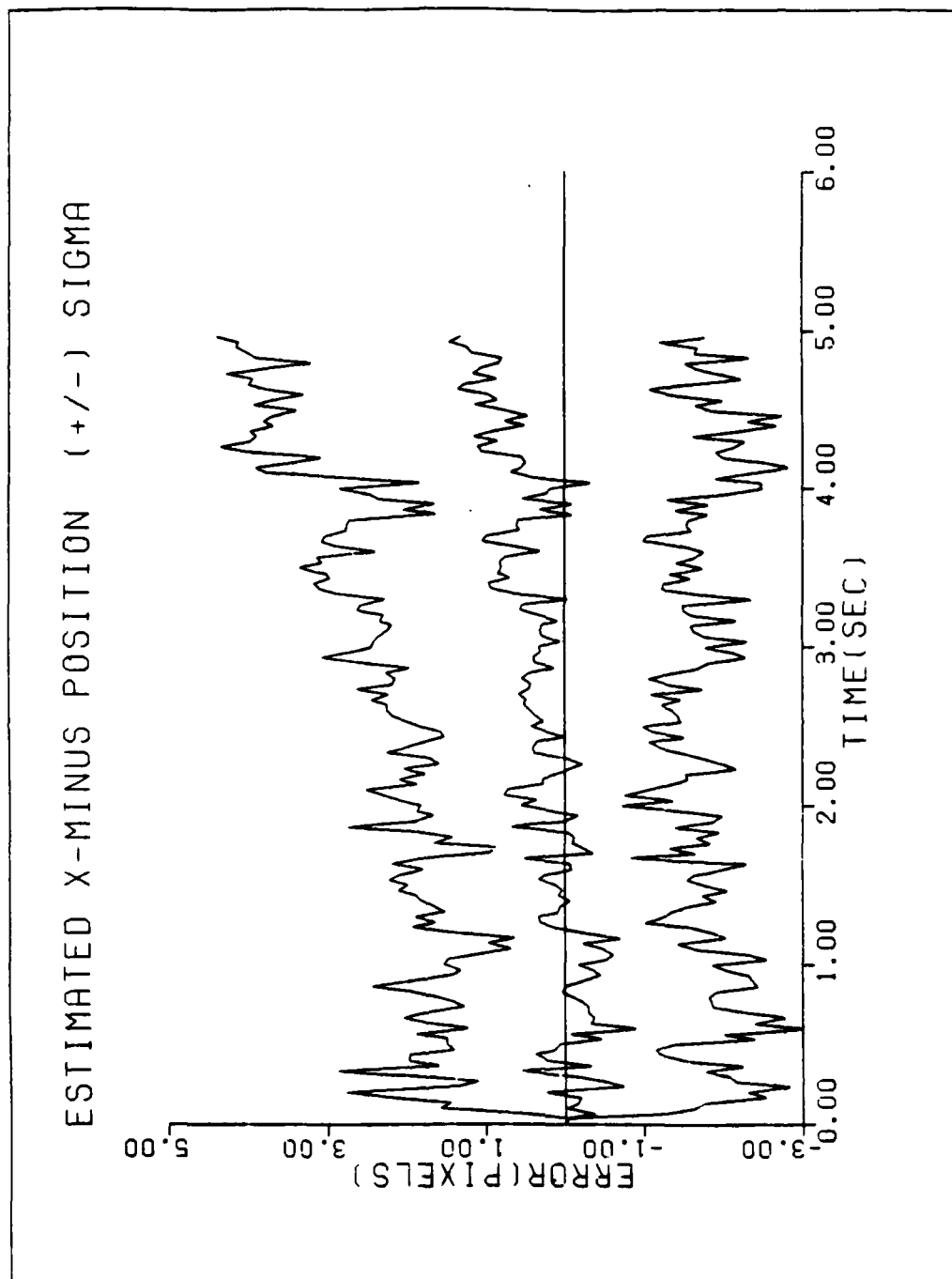


Figure C32 Performance plots for T2G10S/N1.25

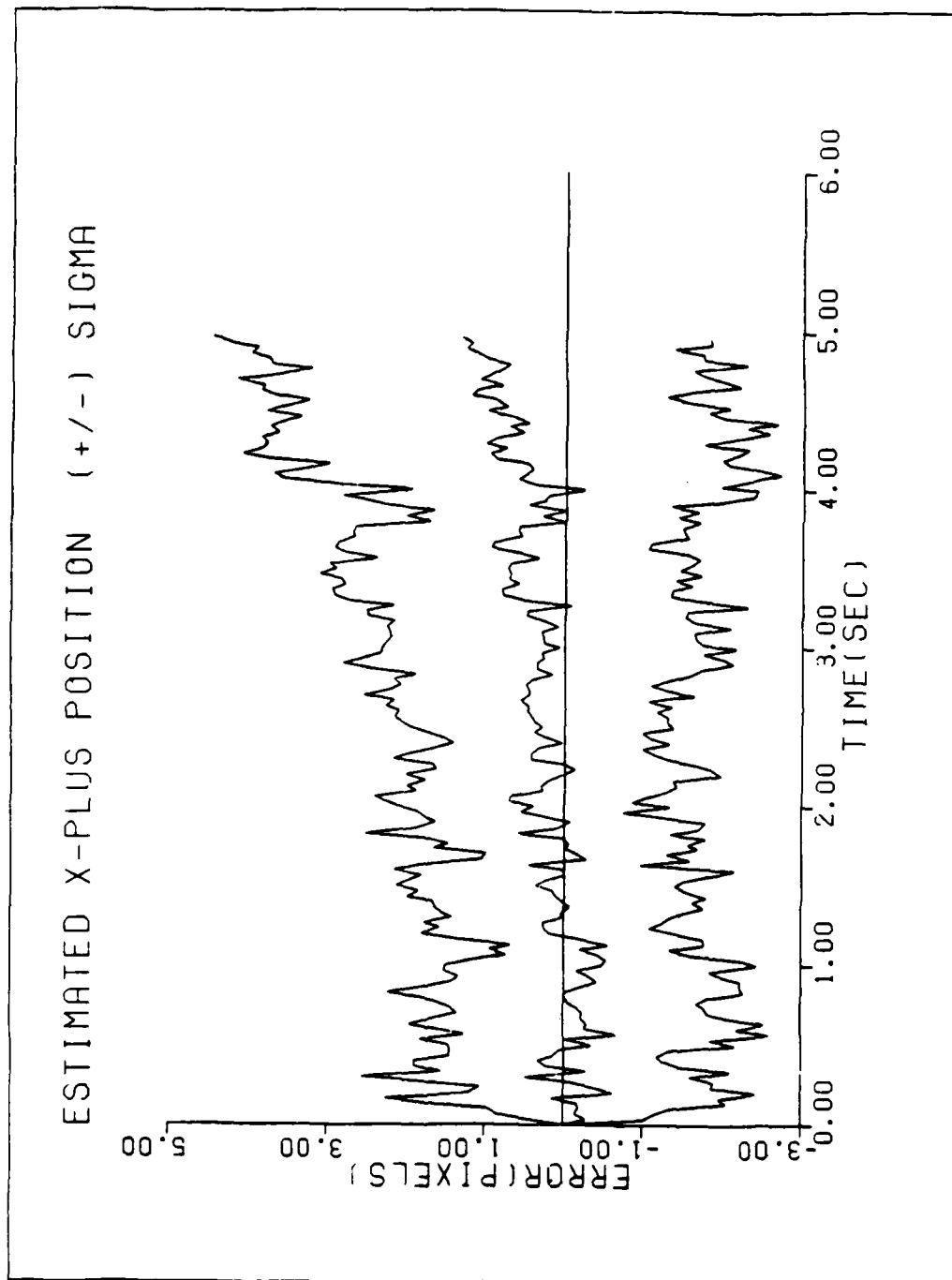


Figure C33 Performance plots for T2G10S/N1.25

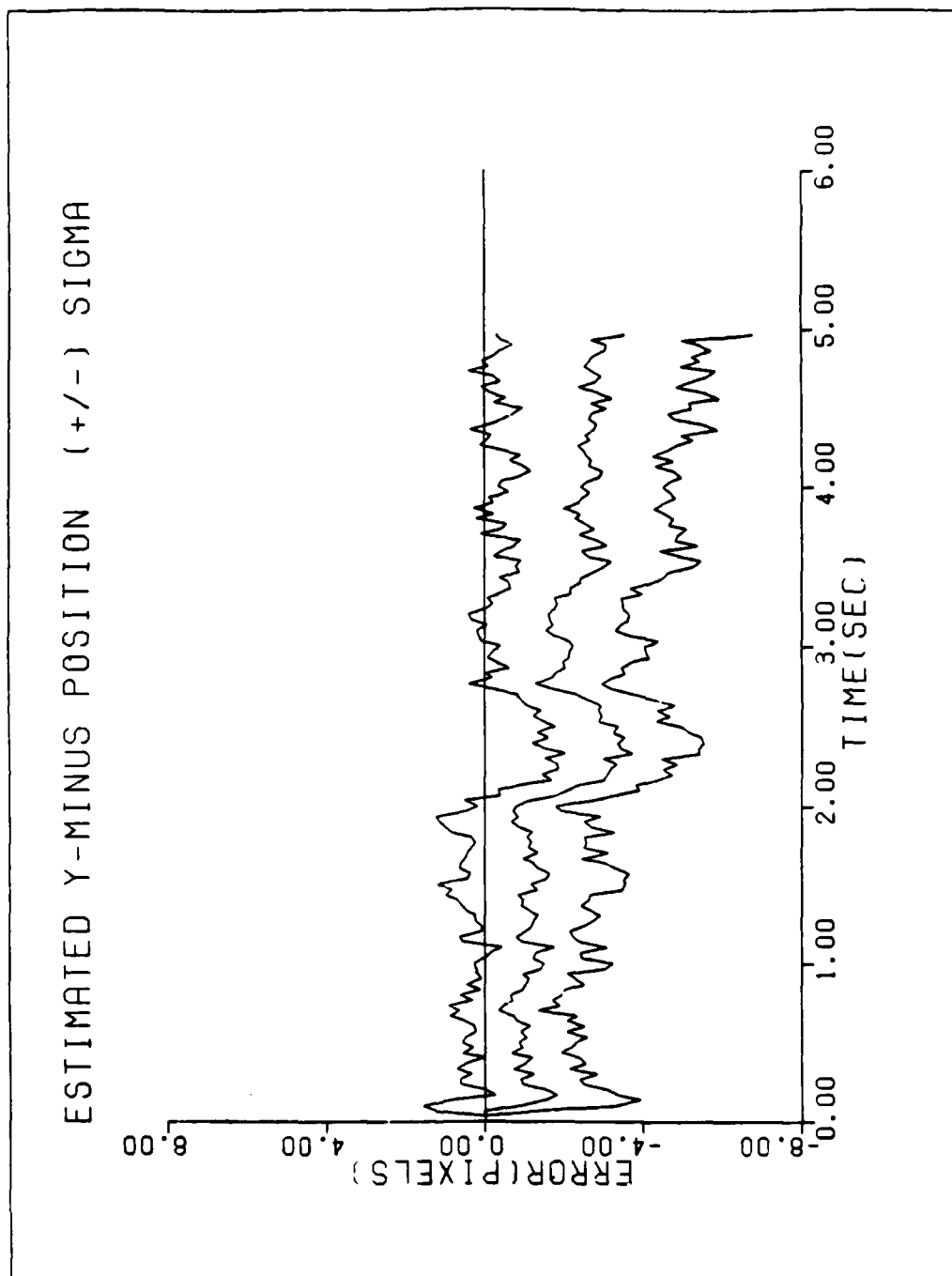


Figure C34 Performance plots for T2G10S/N1.25

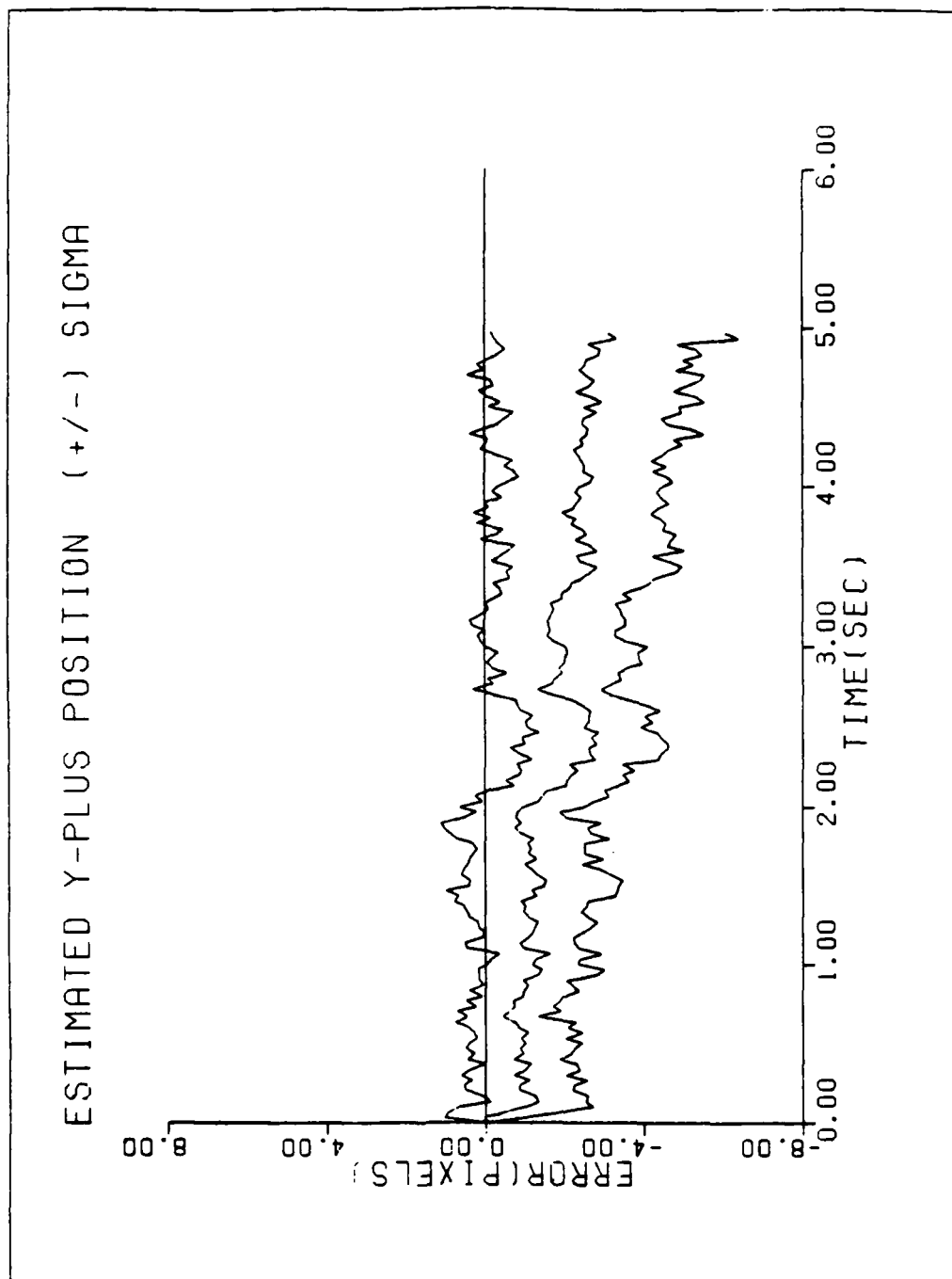


Figure C35 Performance plots for T2G10S/N1.25

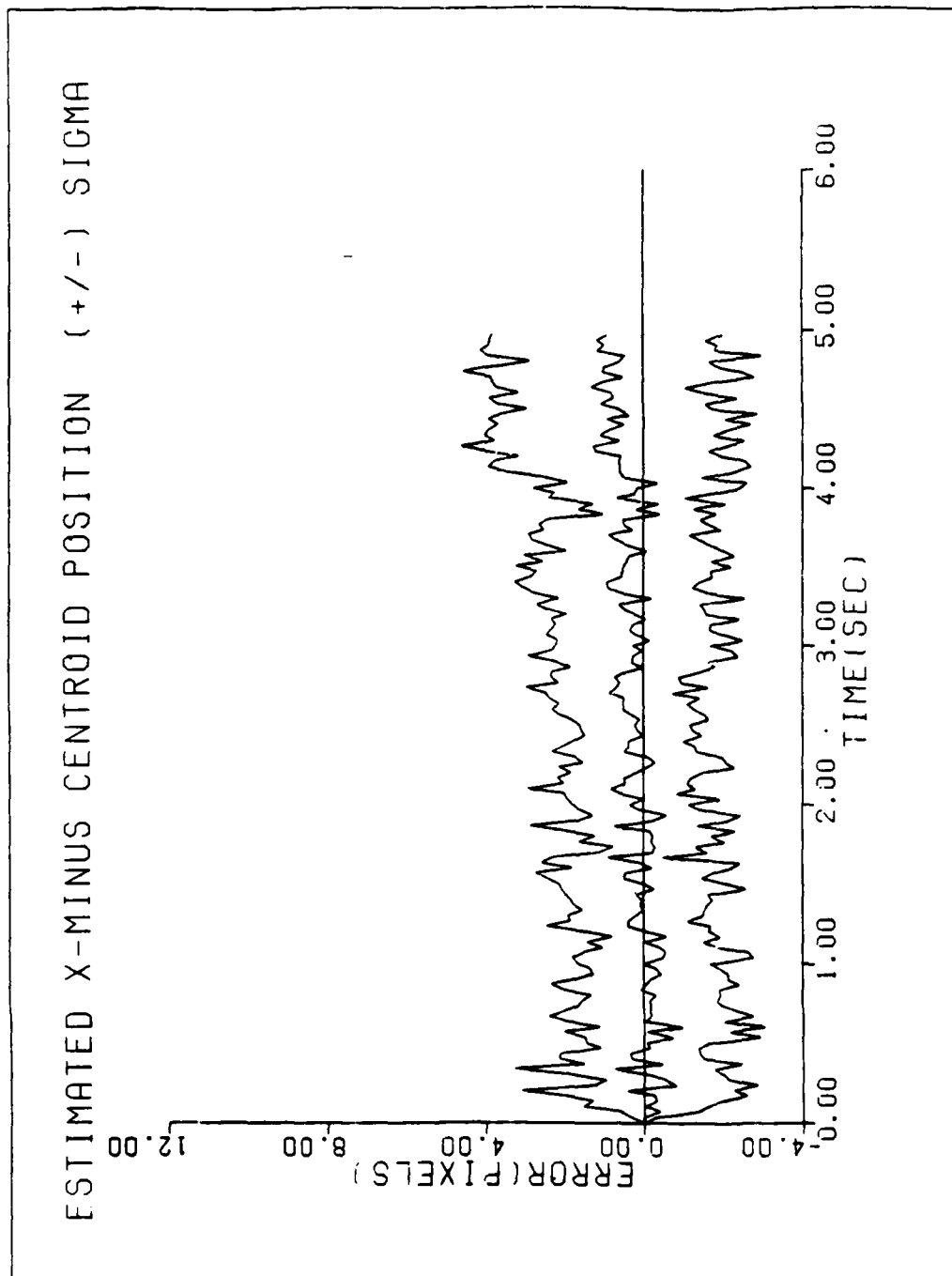


Figure C36 Performance plots for T2G10S/N1.25

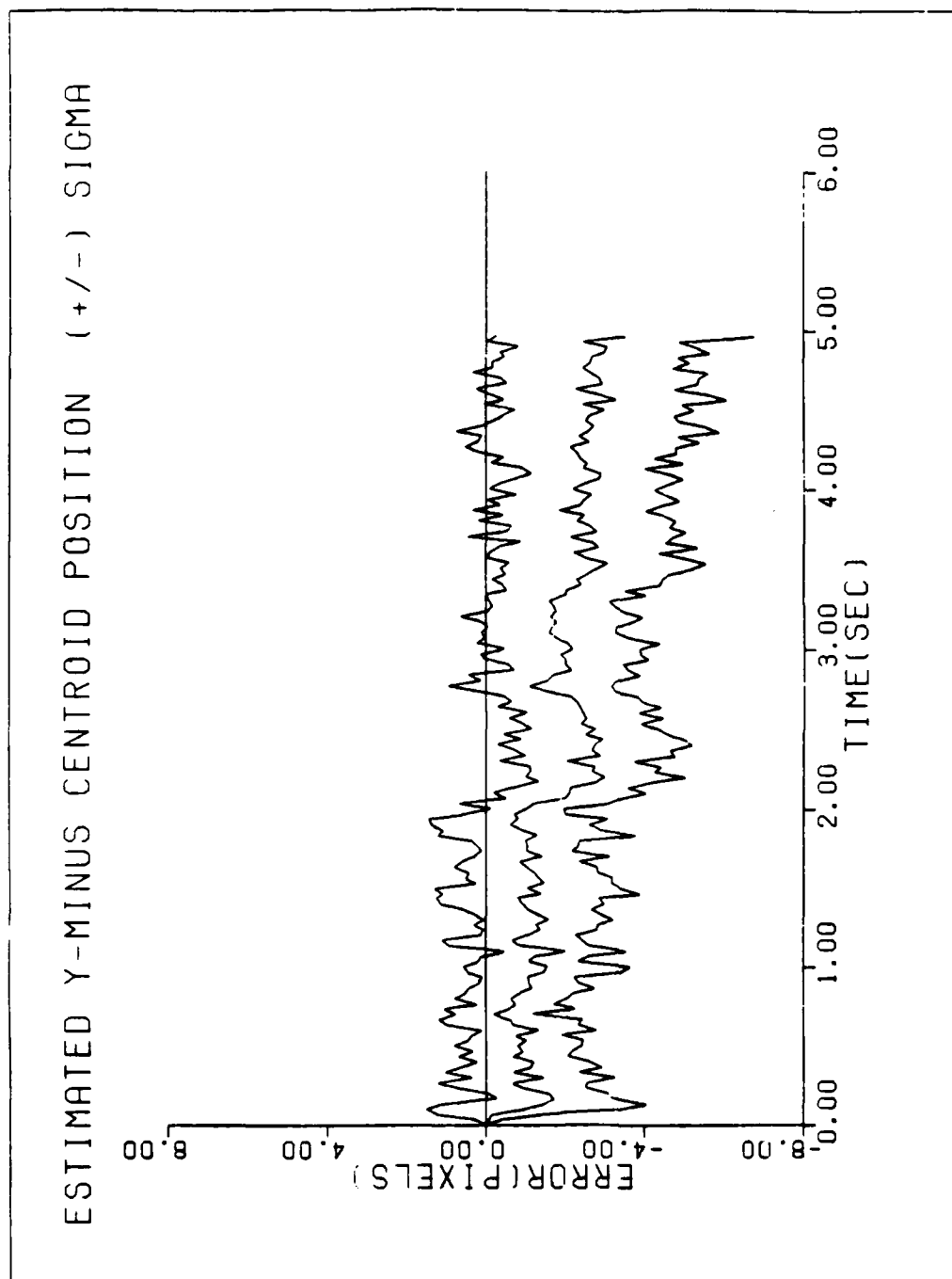


Figure C37 Performance plots for T2G10S/N1.25

APPENDIX D
TARGET RANGE FROM SENSOR / PIXEL SIZE SENSITIVITY

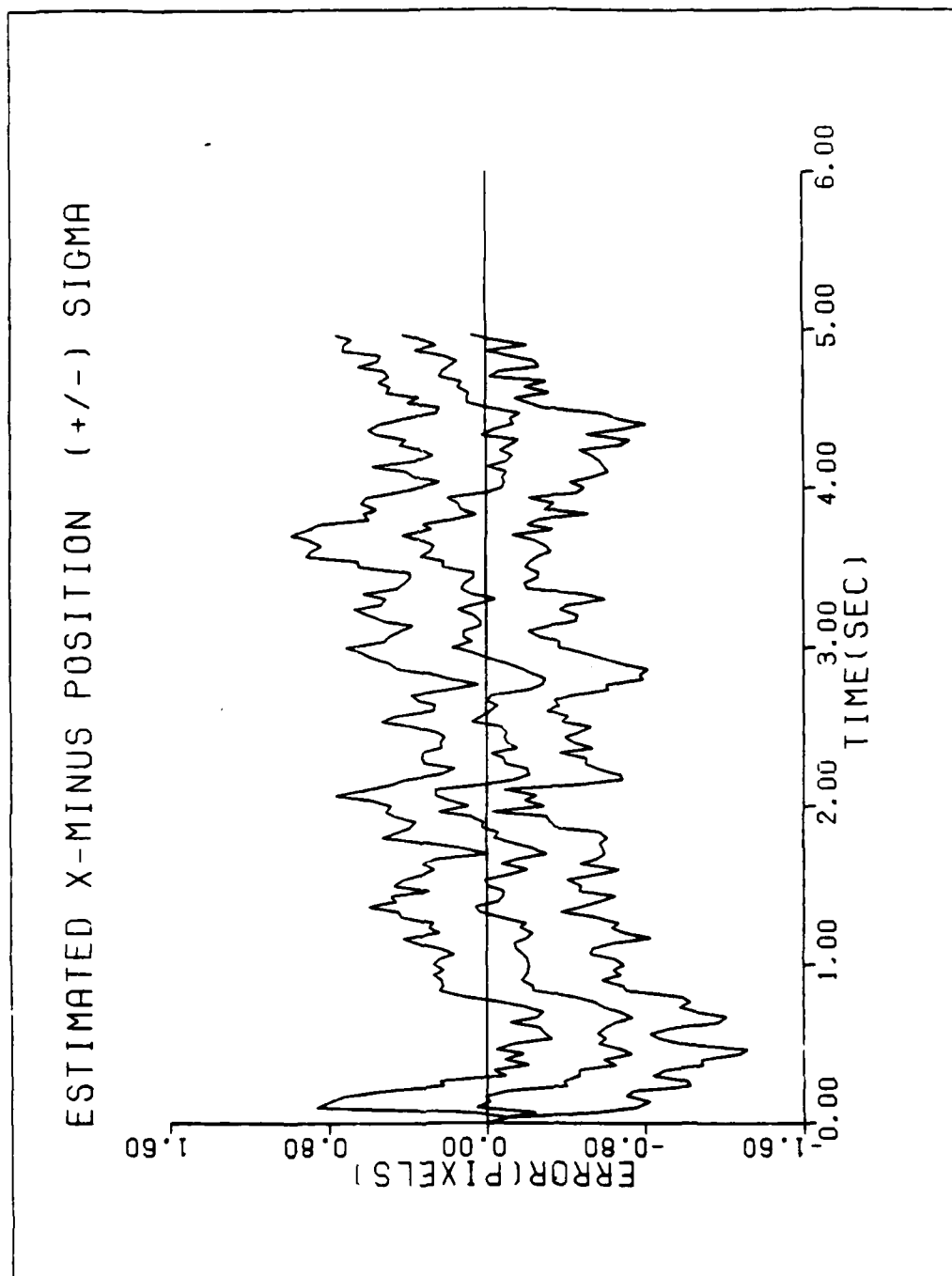


Figure D1 Performance plots for T1Z015IC

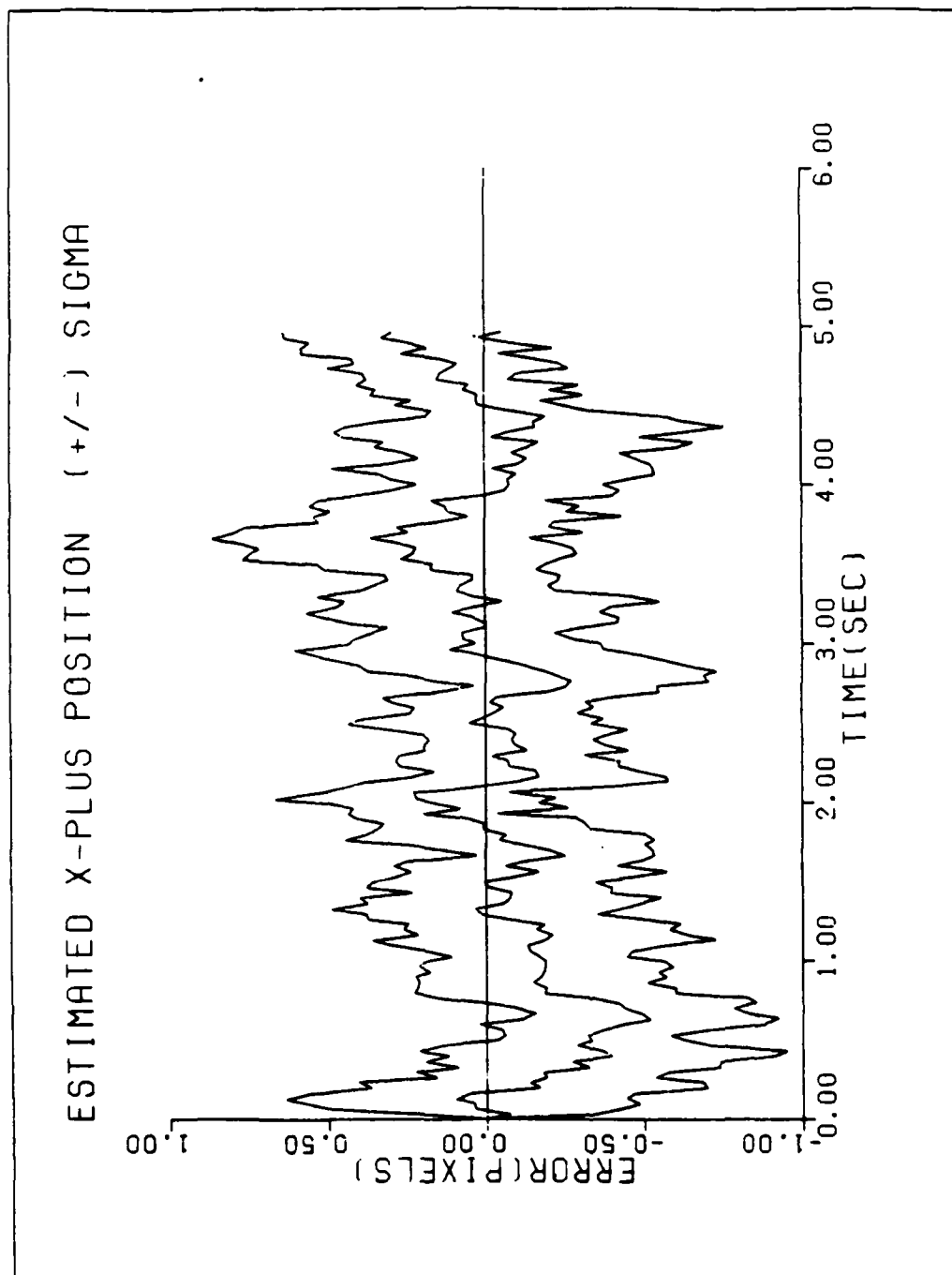


Figure D2 Performance plots for T1Z015IC

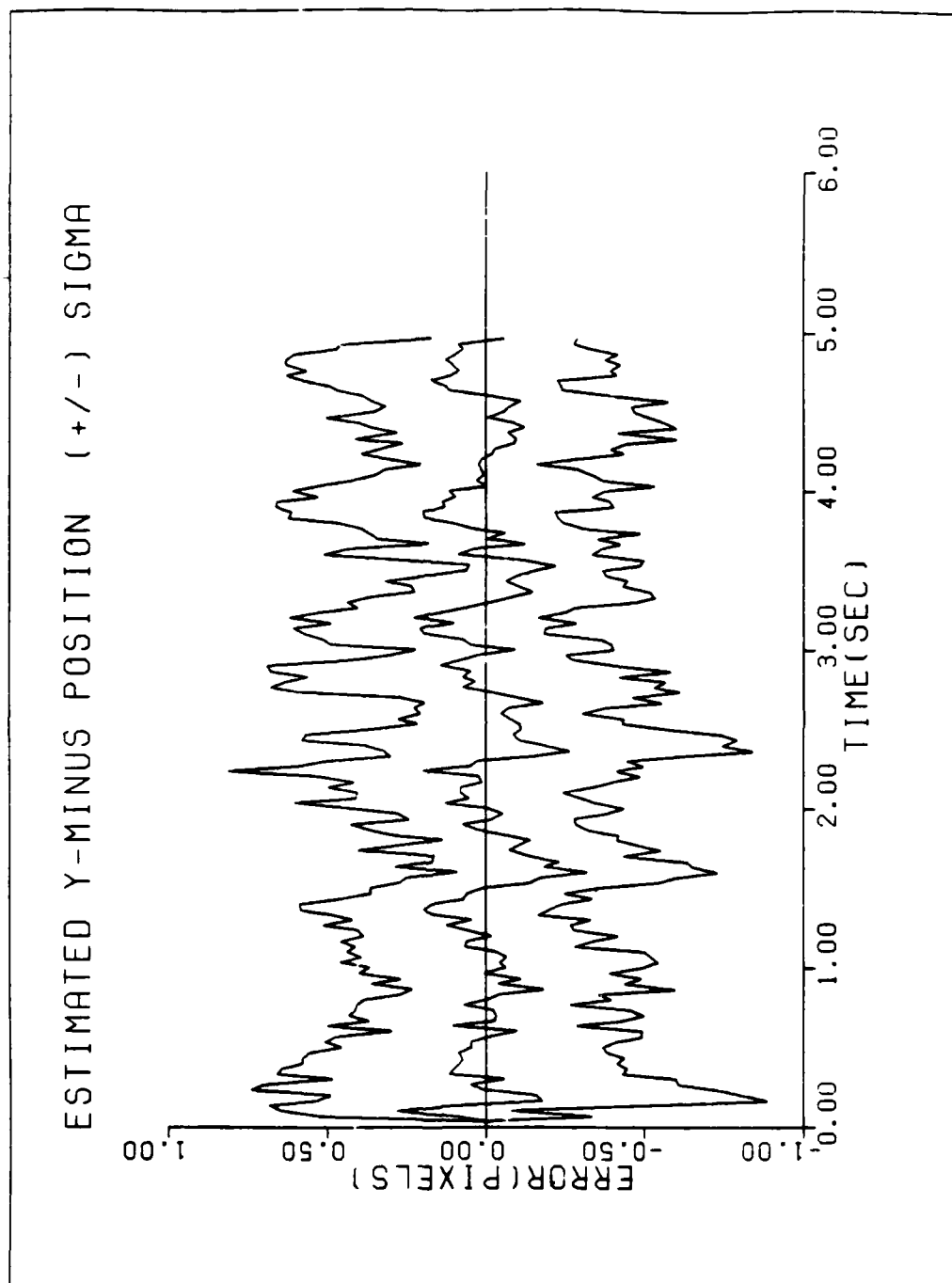


Figure D3 Performance plots for T1Z015IC

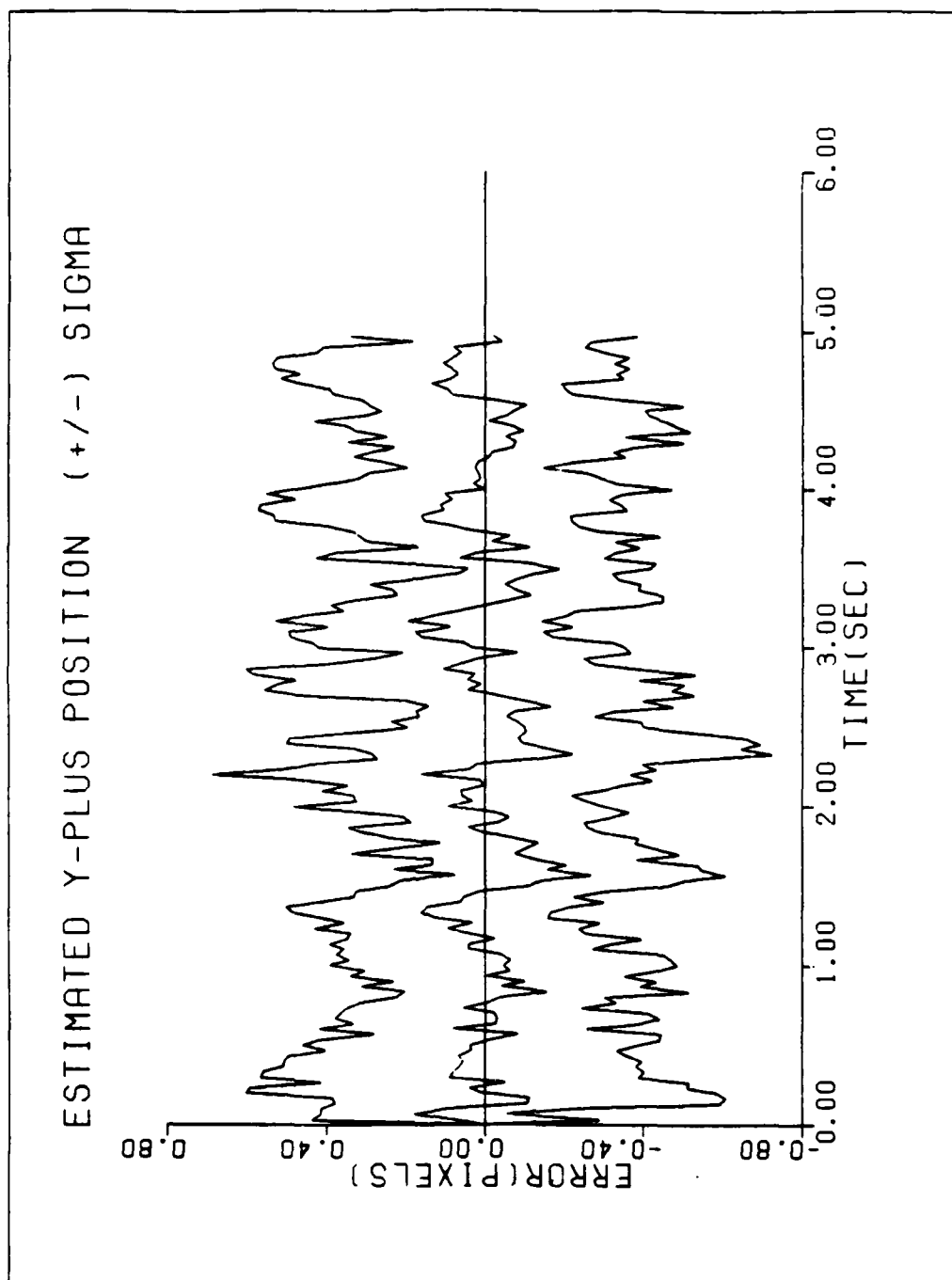


Figure D4 Performance plots for T1Z015IC

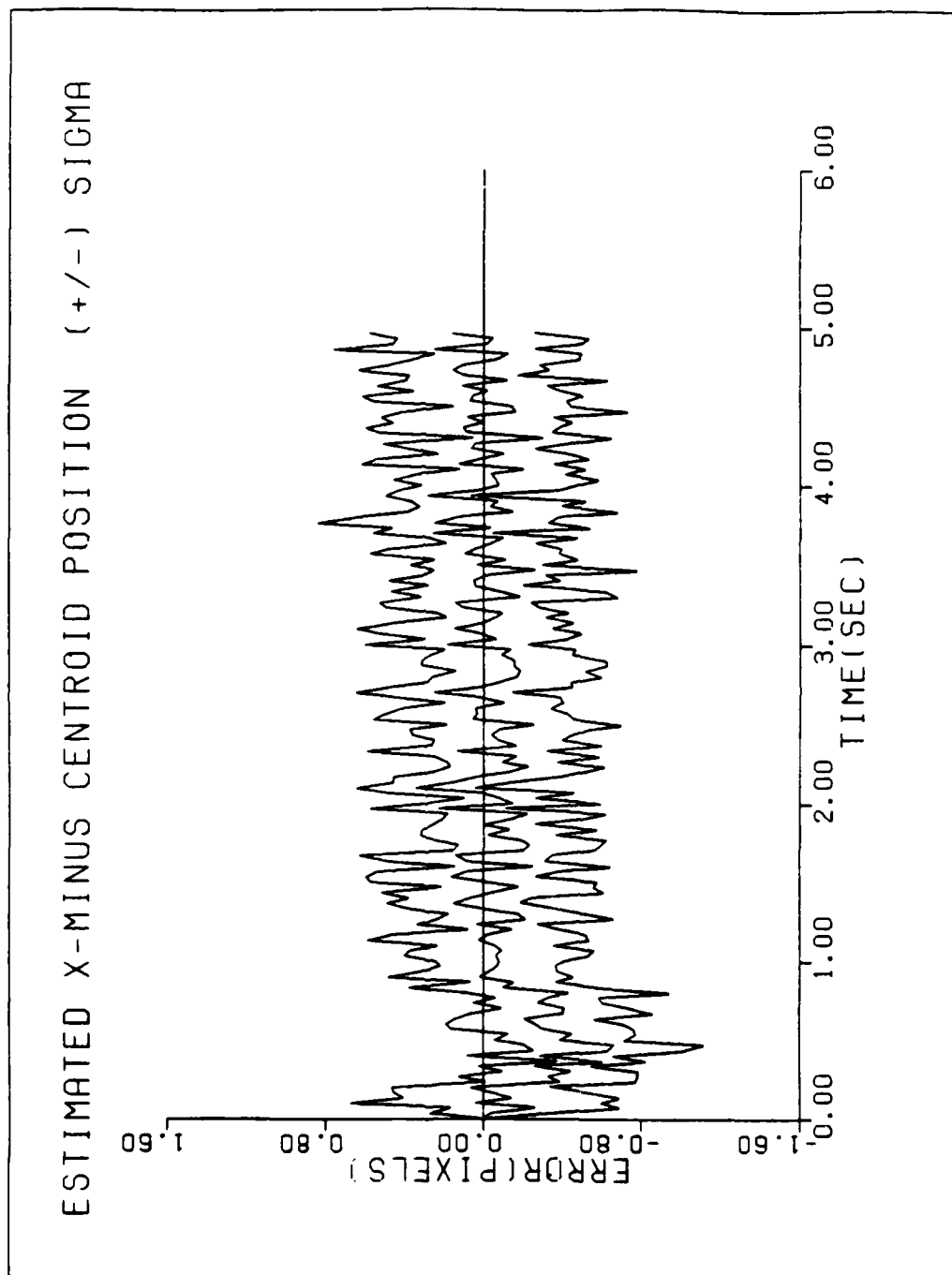


Figure D5 Performance plots for T1Z015IC

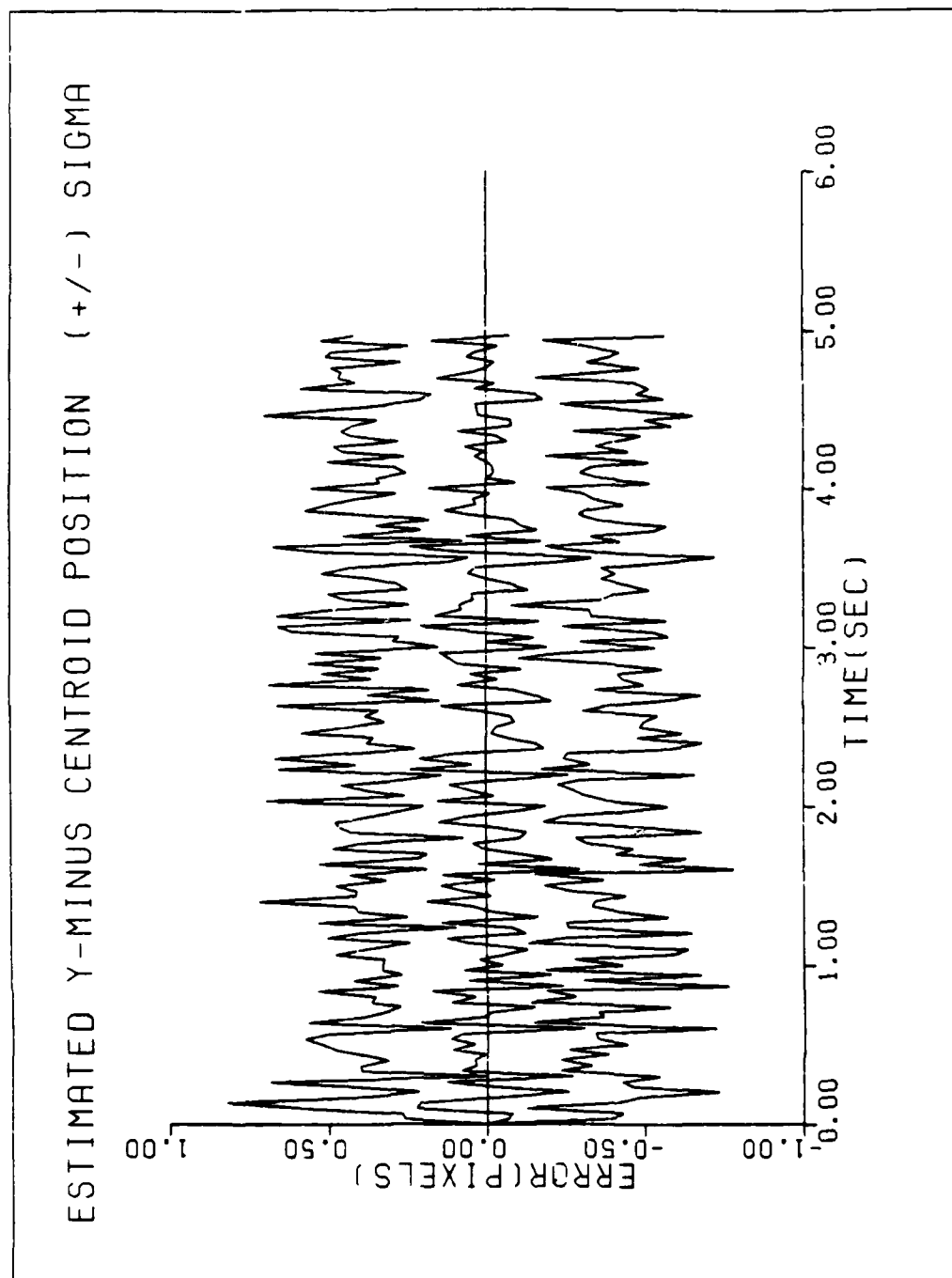


Figure D6 Performance plots for T1Z015IC

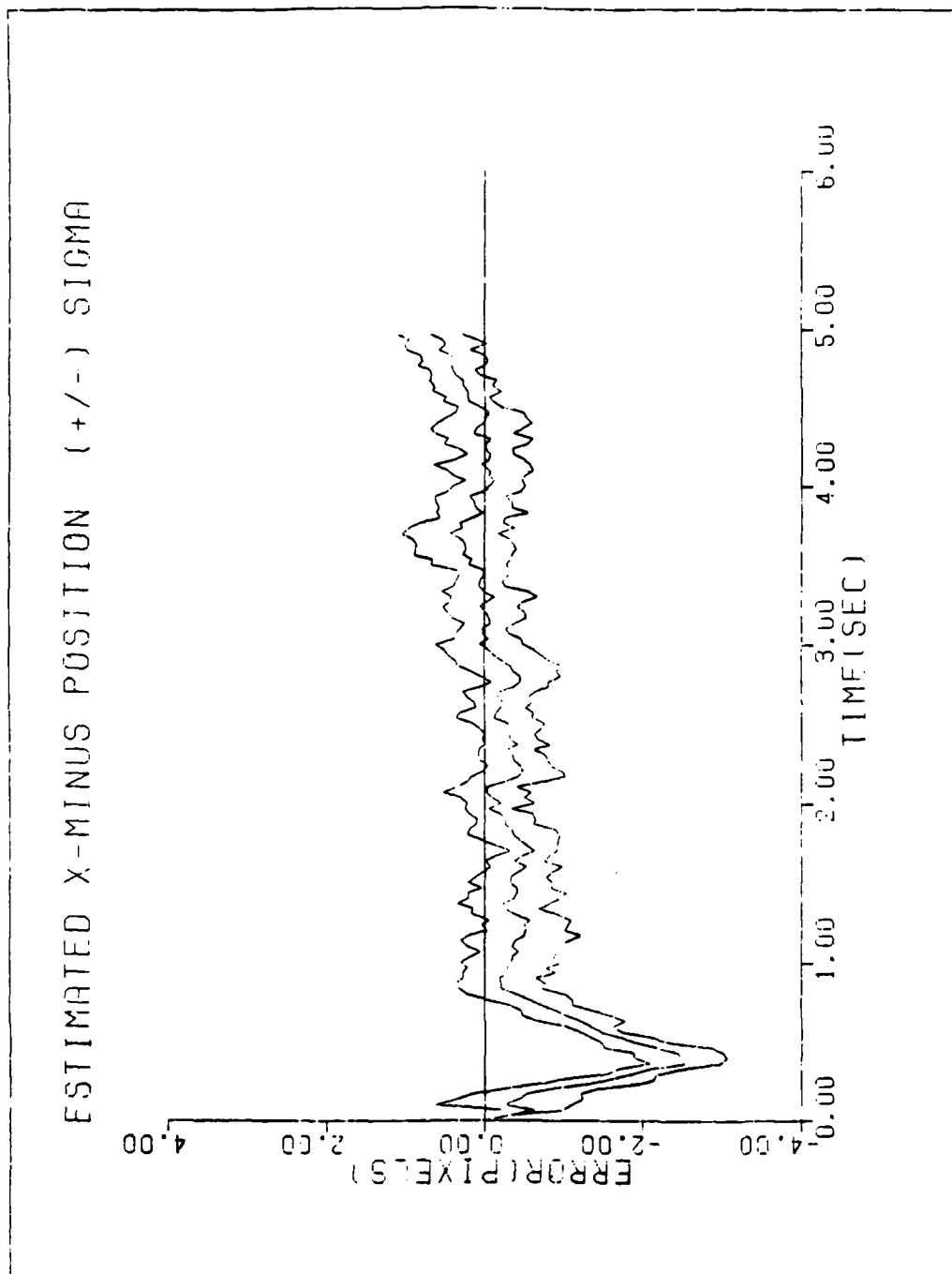


Figure D7 Performance plots for T1Z015IC

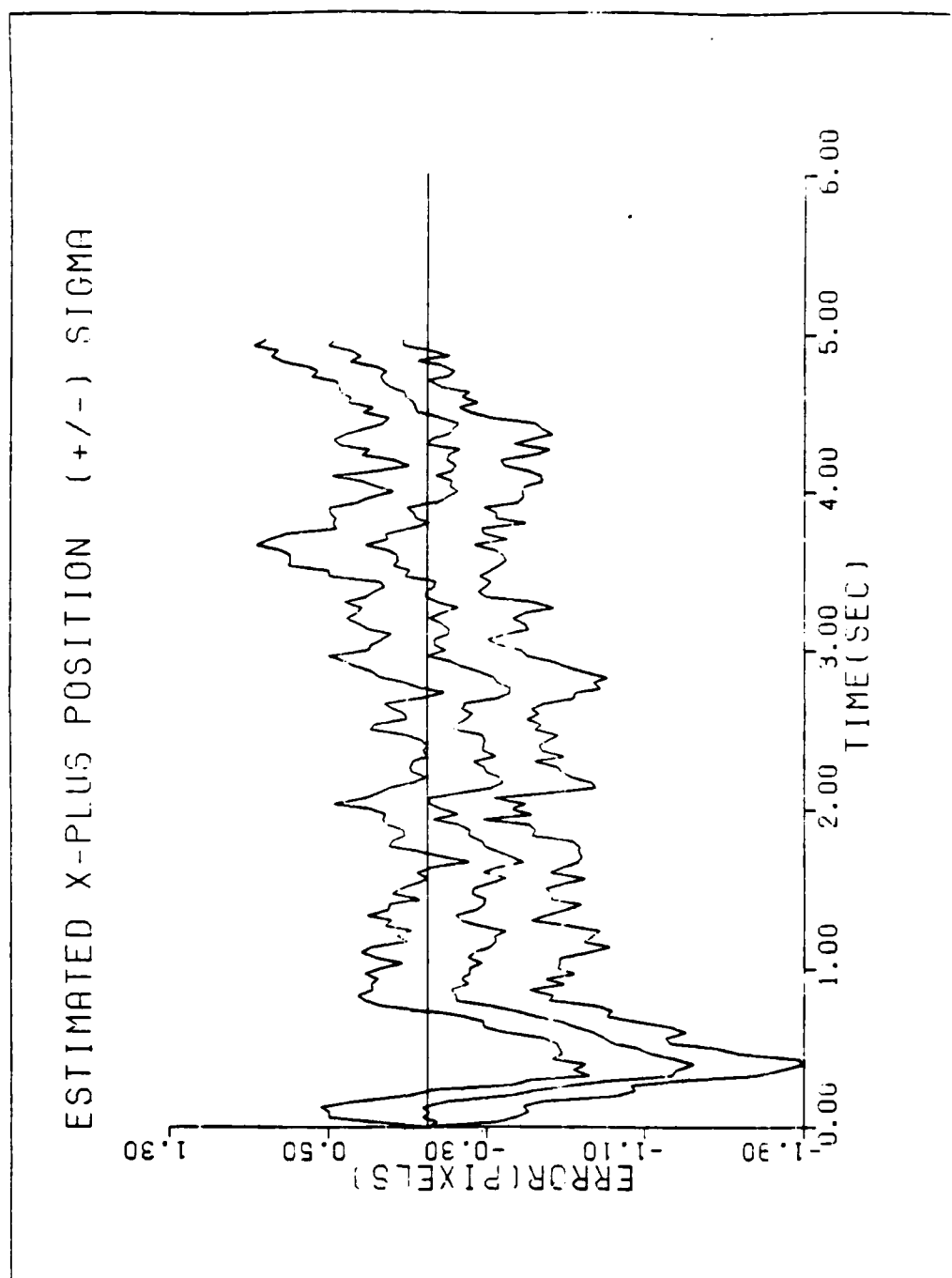


Figure D8 Performance plots for T1Z015IC

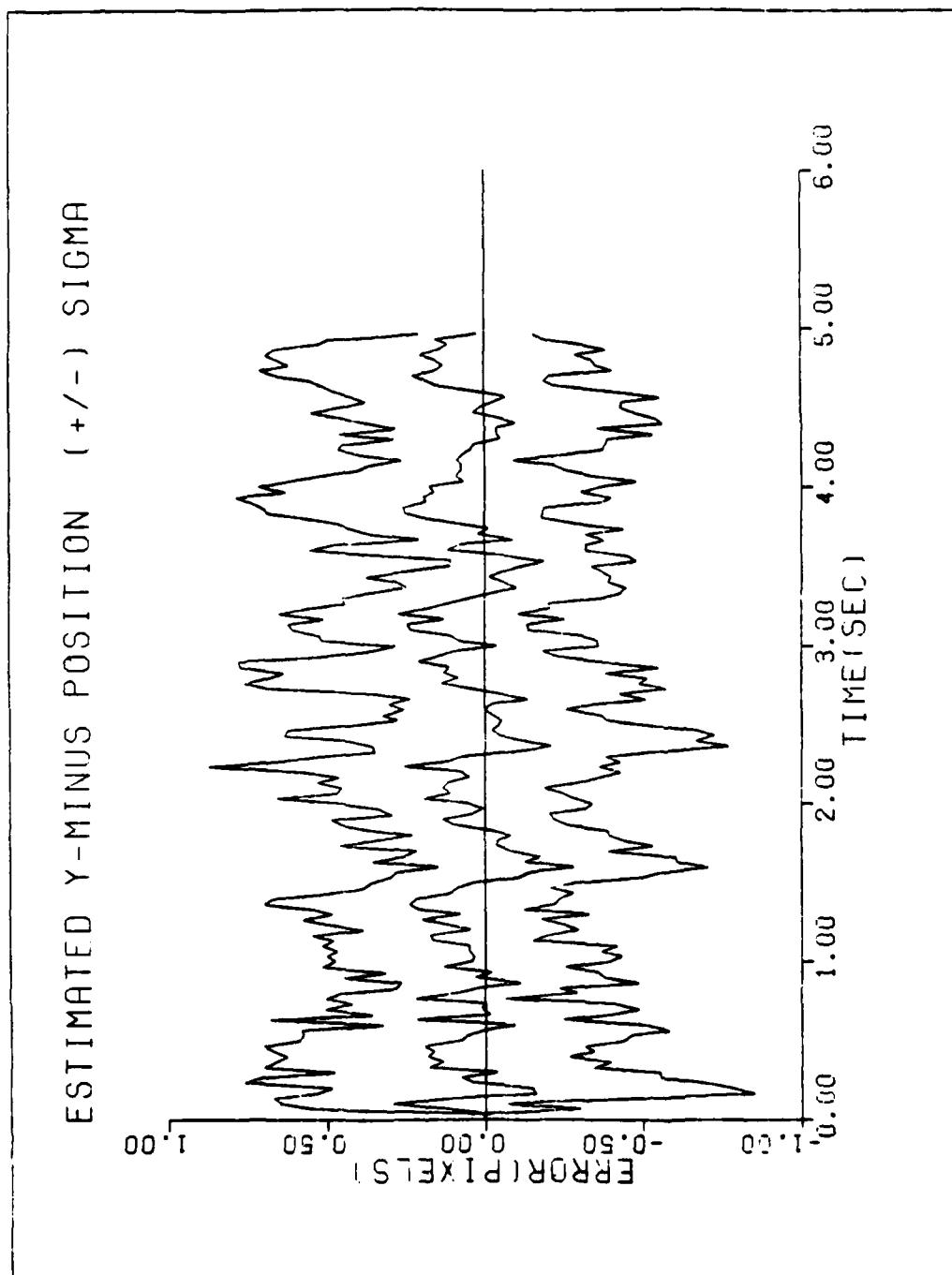


Figure D9 Performance plots for T1Z015IC

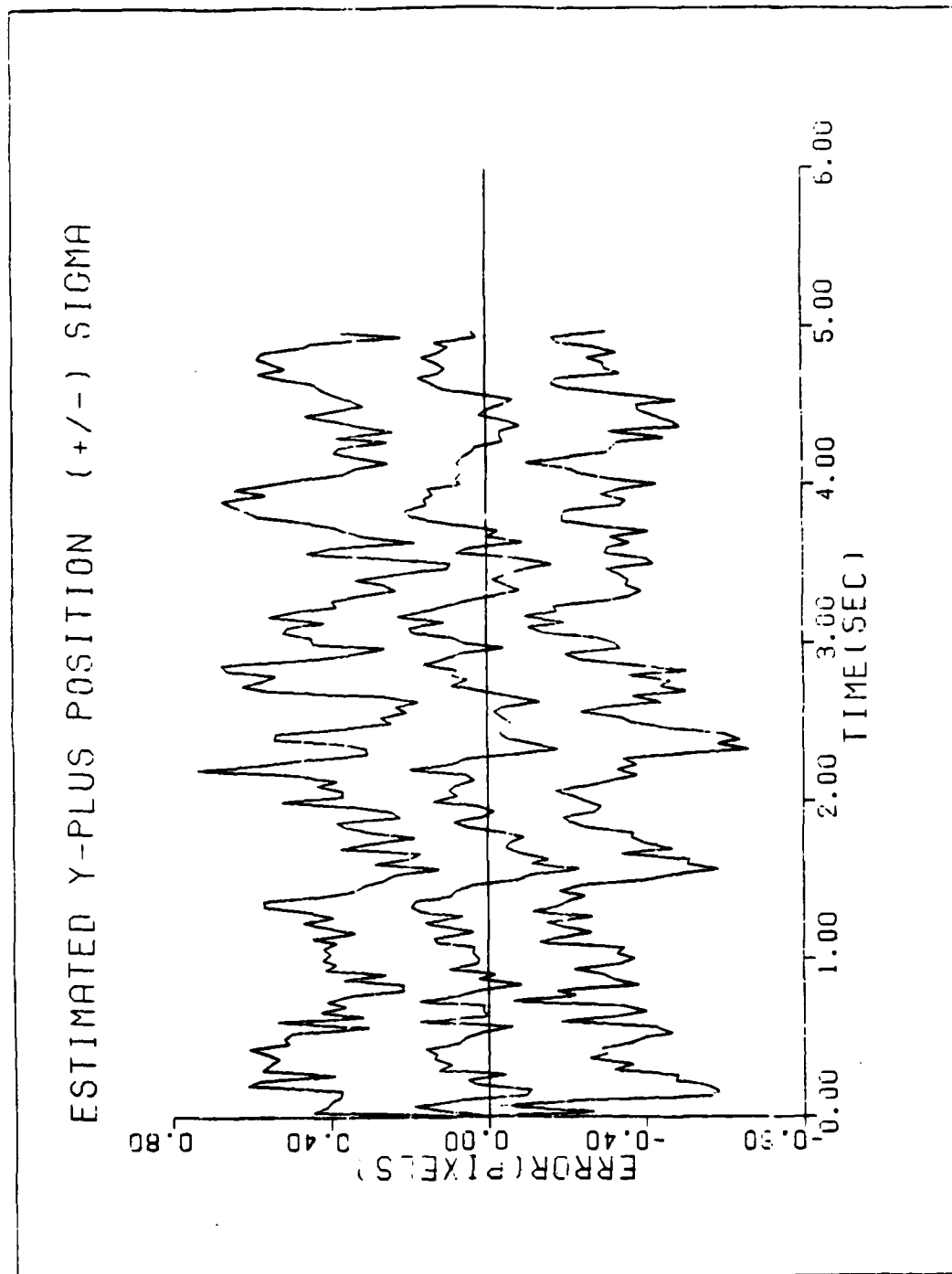


Figure D10 Performance plots for T1Z105IC

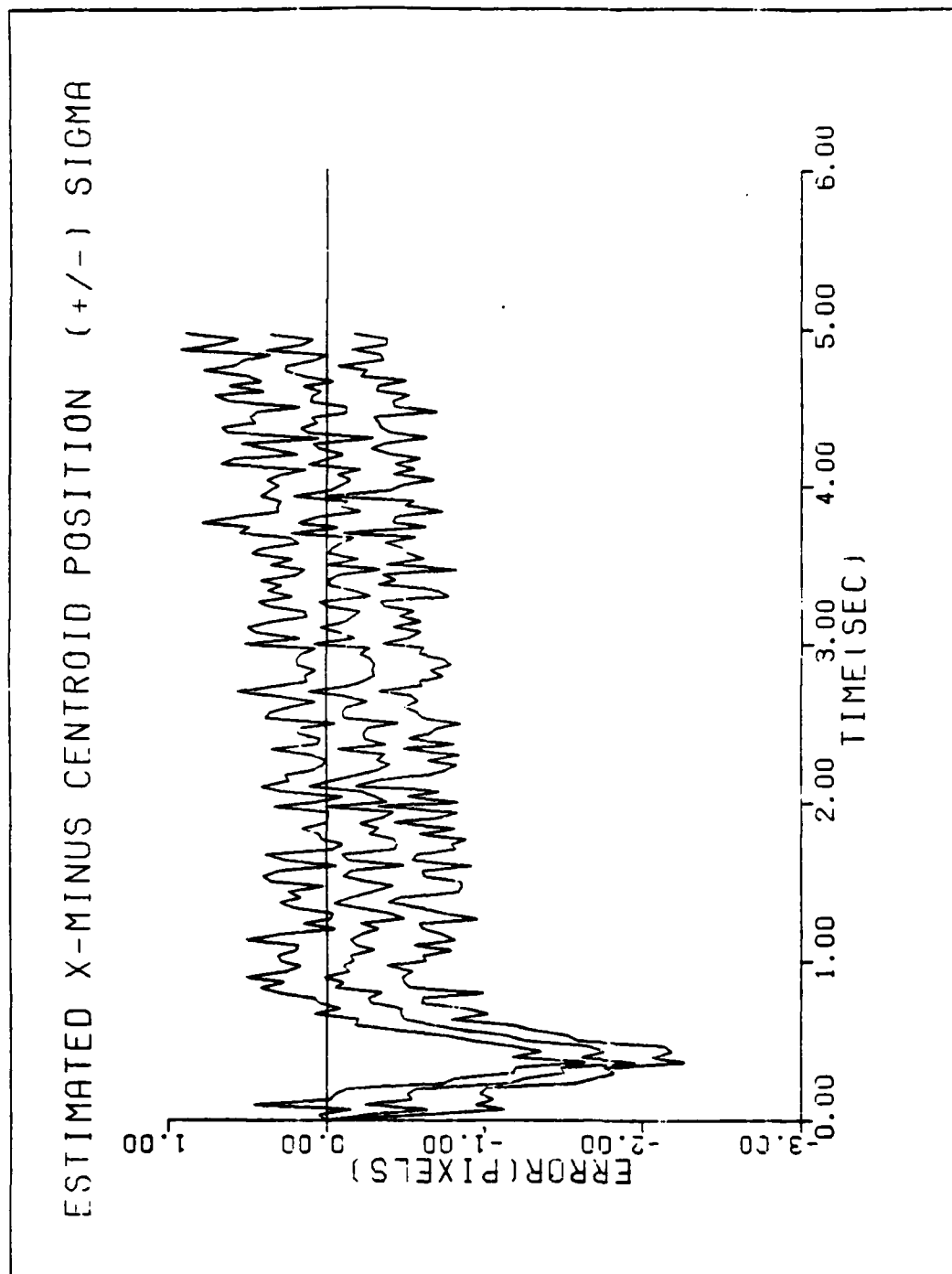


Figure D11 Performance plots for T1Z015IC

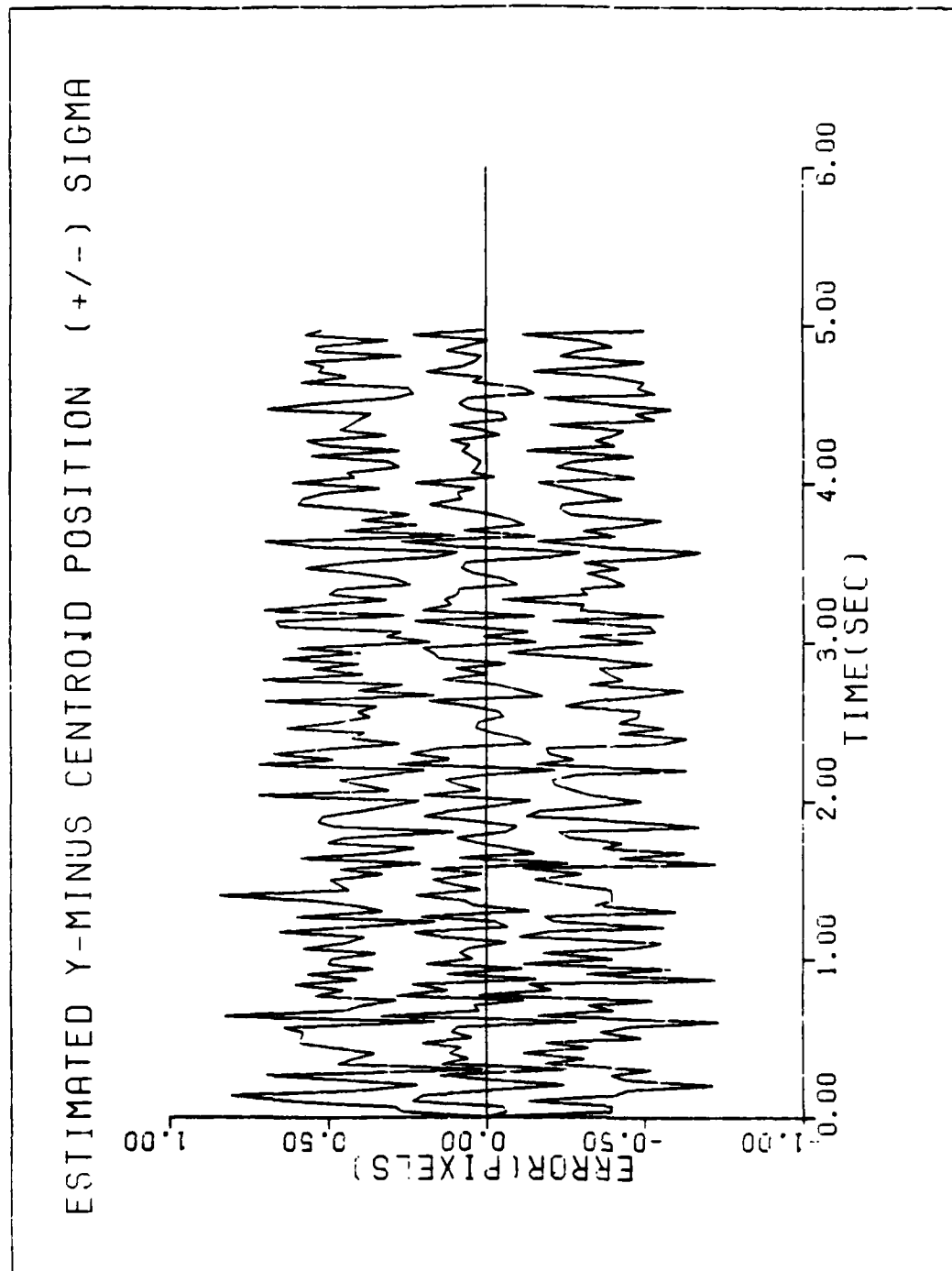


Figure D12 Performance plots for T1Z010IC

ESTIMATED X-MINUS POSITION (+/-) SIGMA

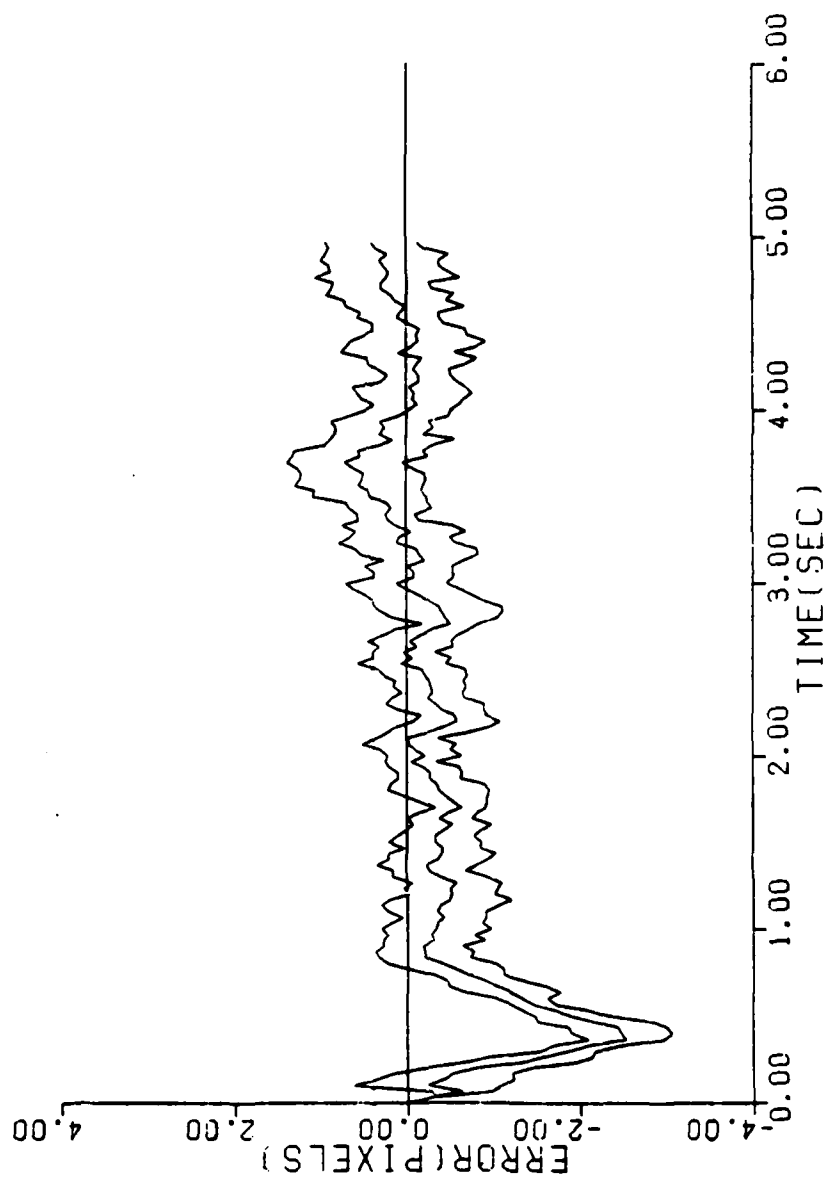


Figure D13 Performance plots for T2G10Z010IC

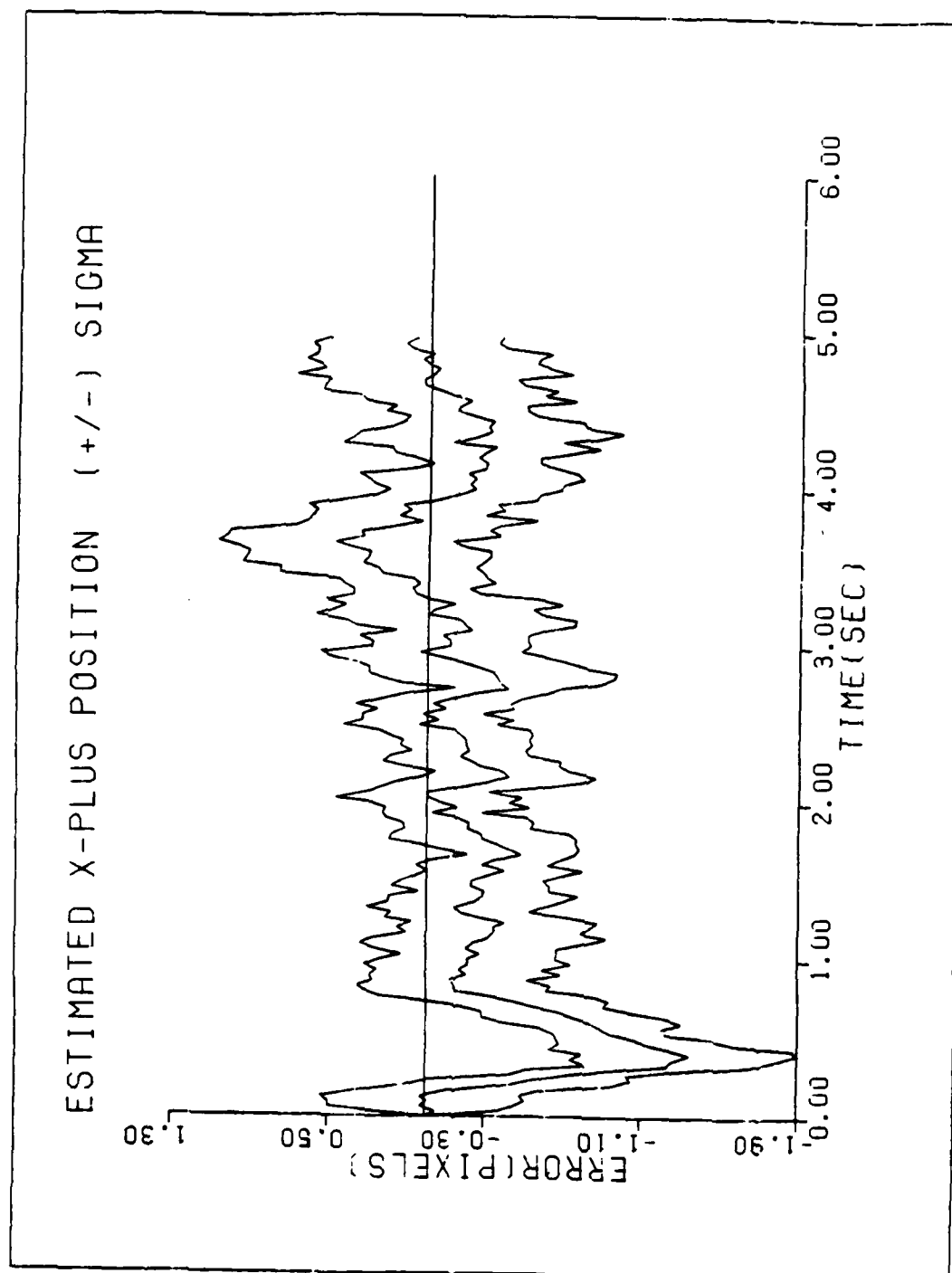


Figure D14 Performance plots for T2G10Z0101C

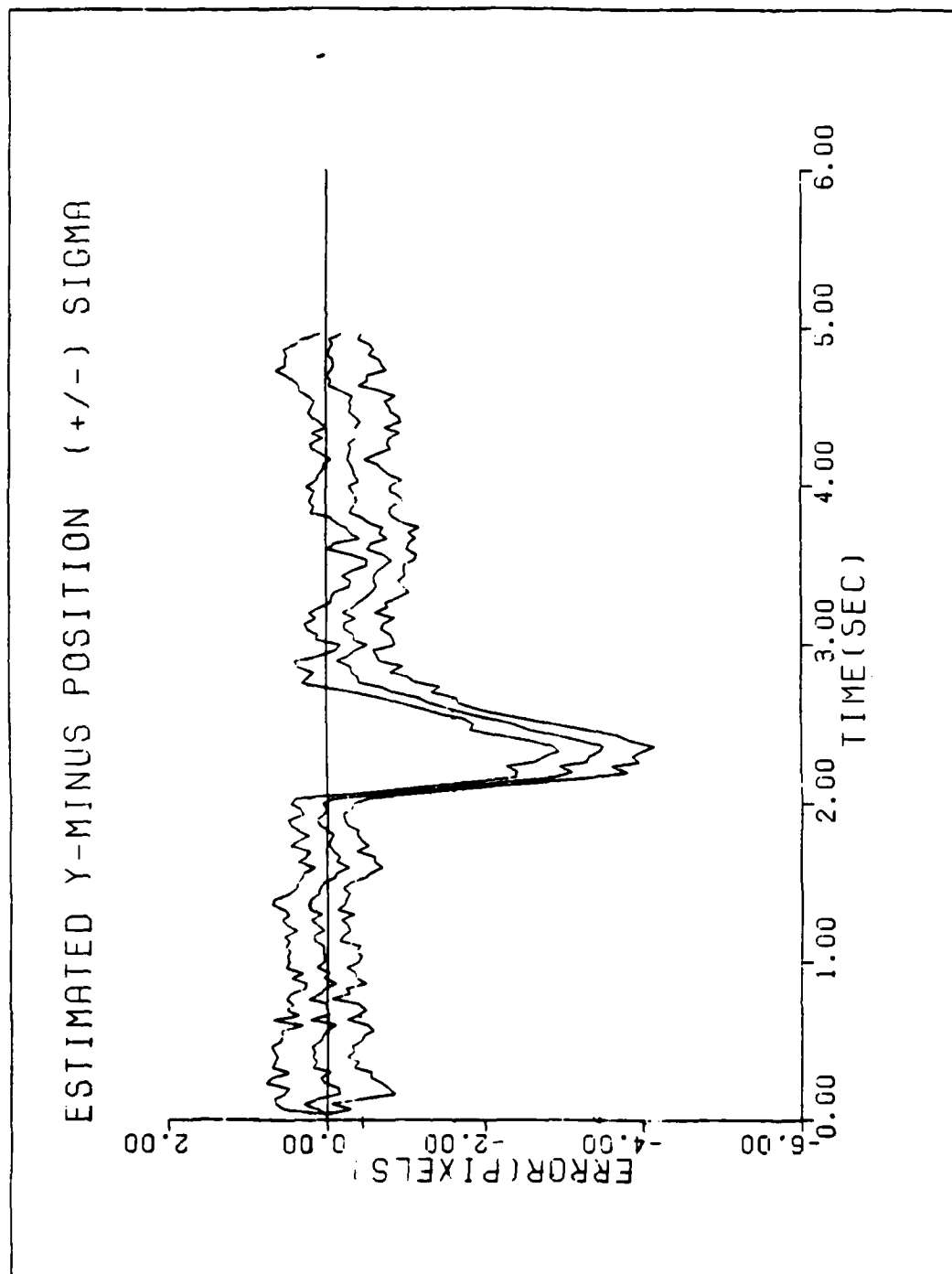


Figure D15 Performance plots for T2G10Z010IC

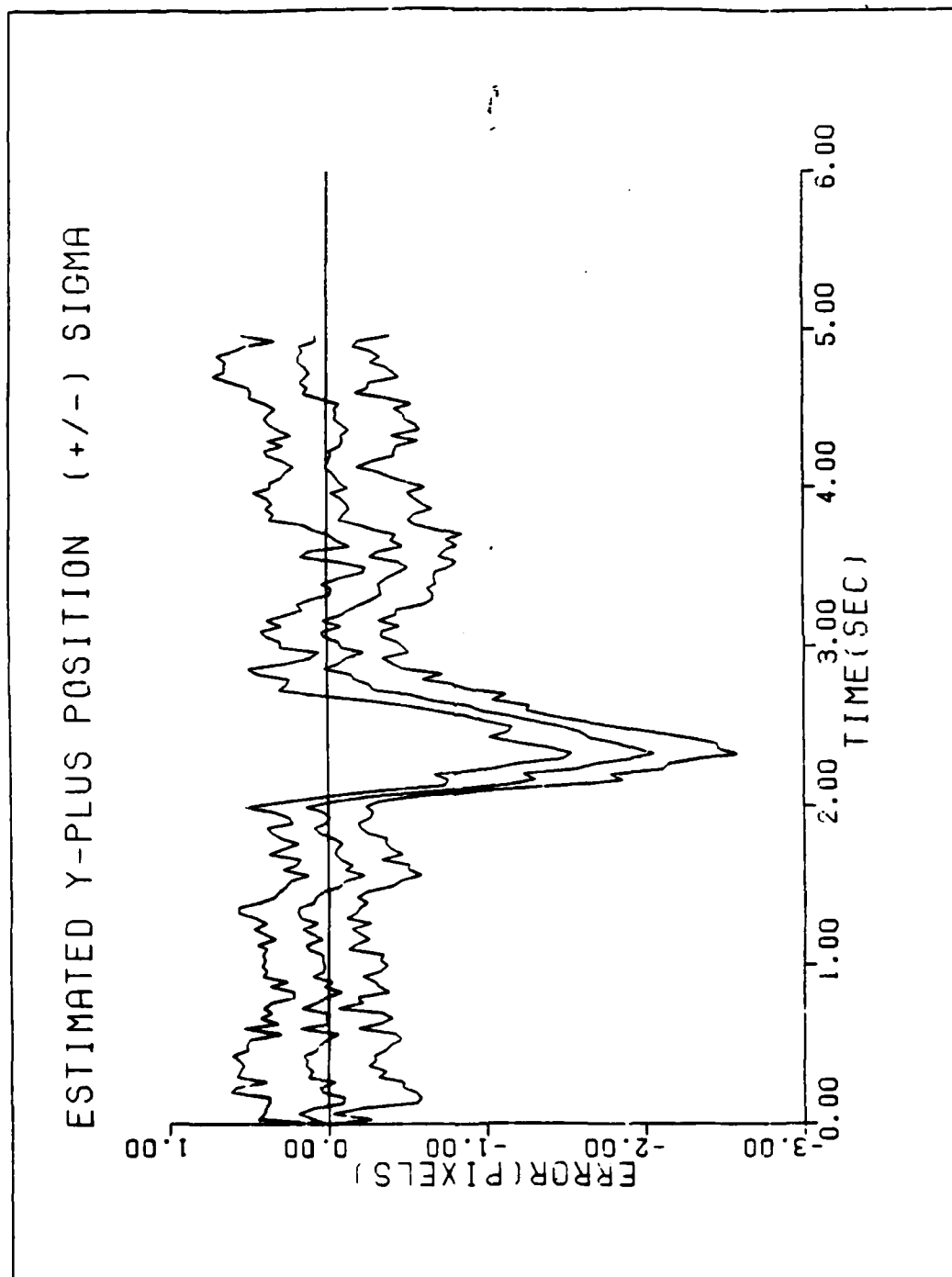


Figure D16 Performance plots for T2G10Z010IC

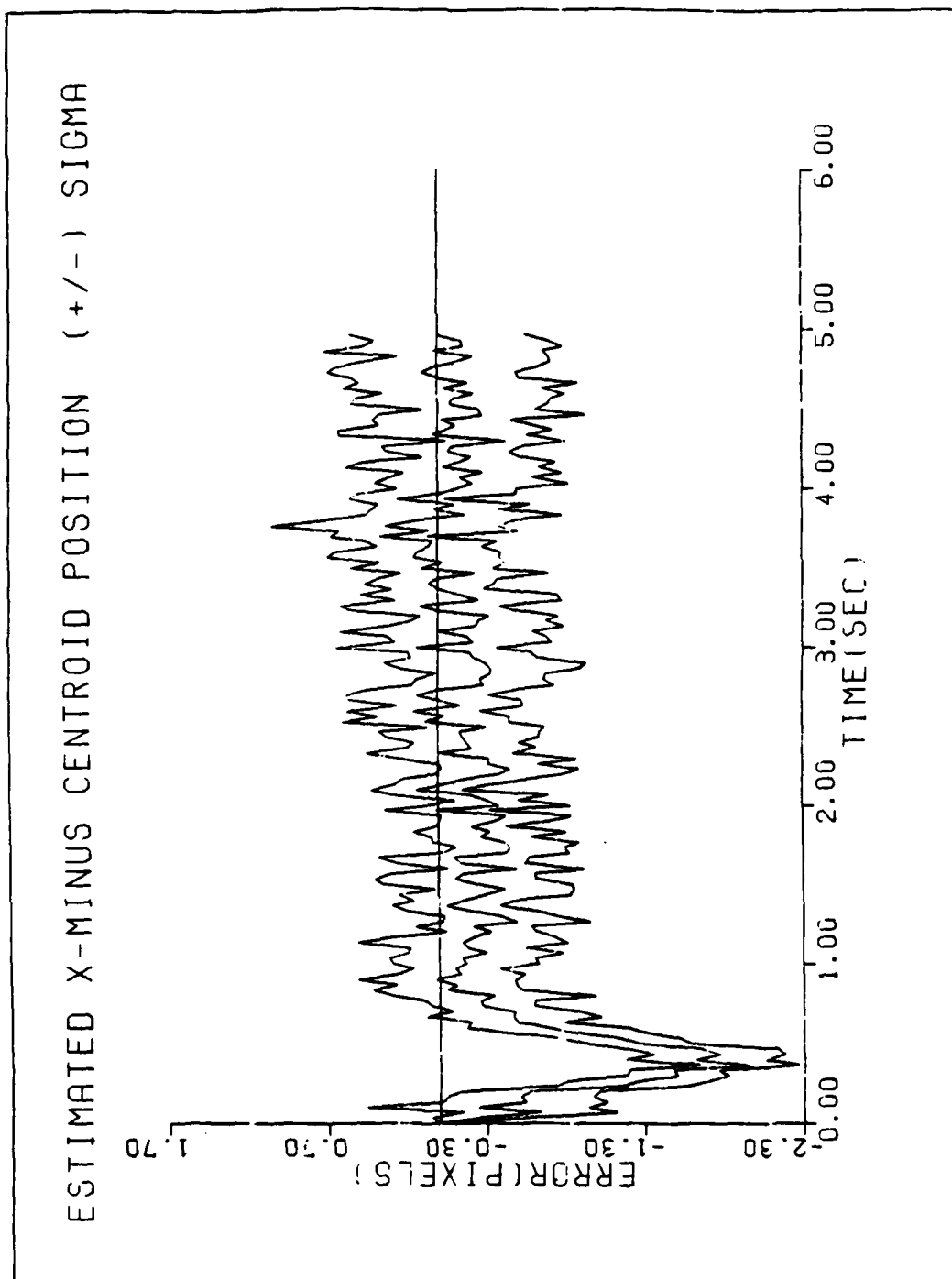


Figure D17 Performance plots for T2G10Z010IC

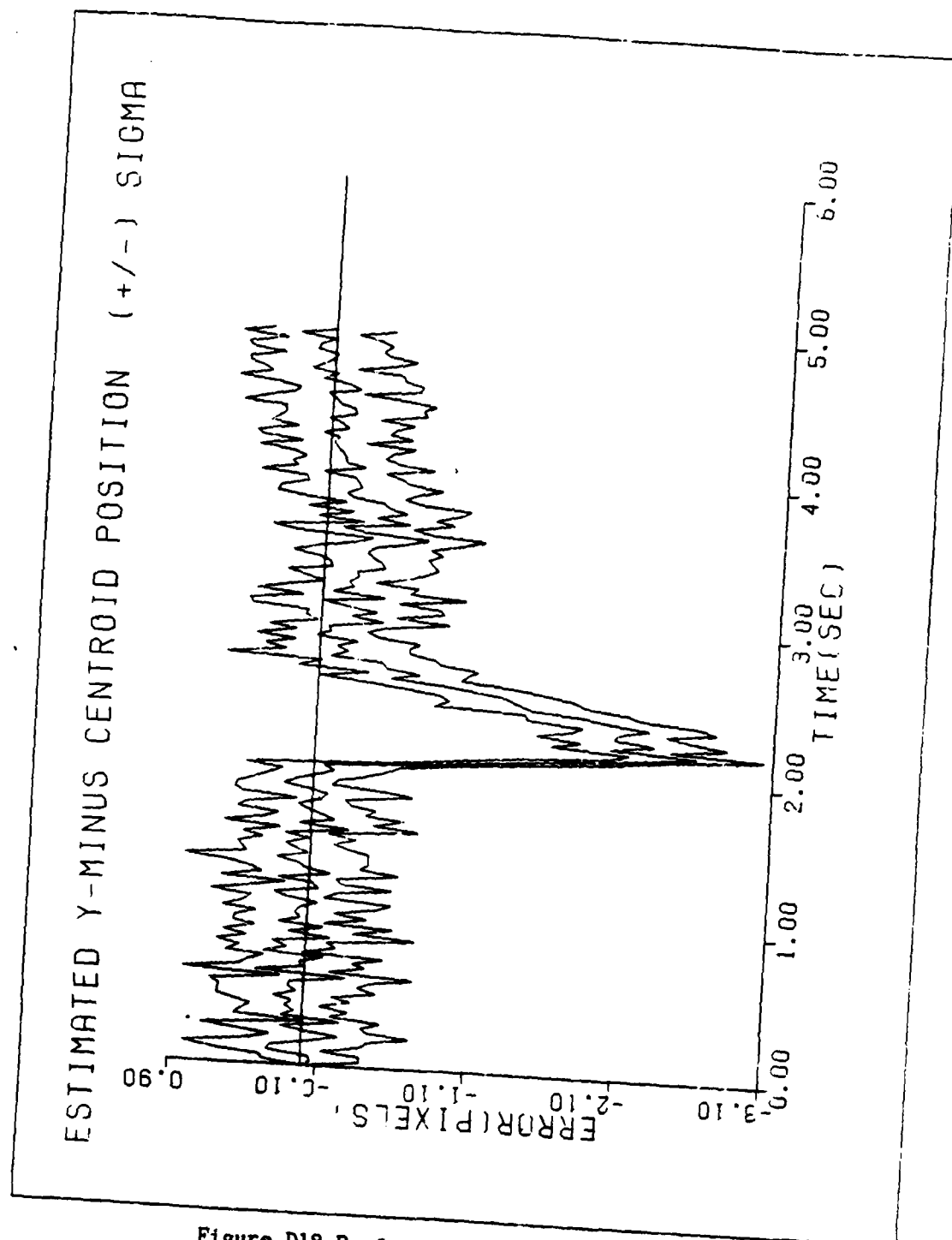


Figure D18 Performance plots for T2G10Z010IC

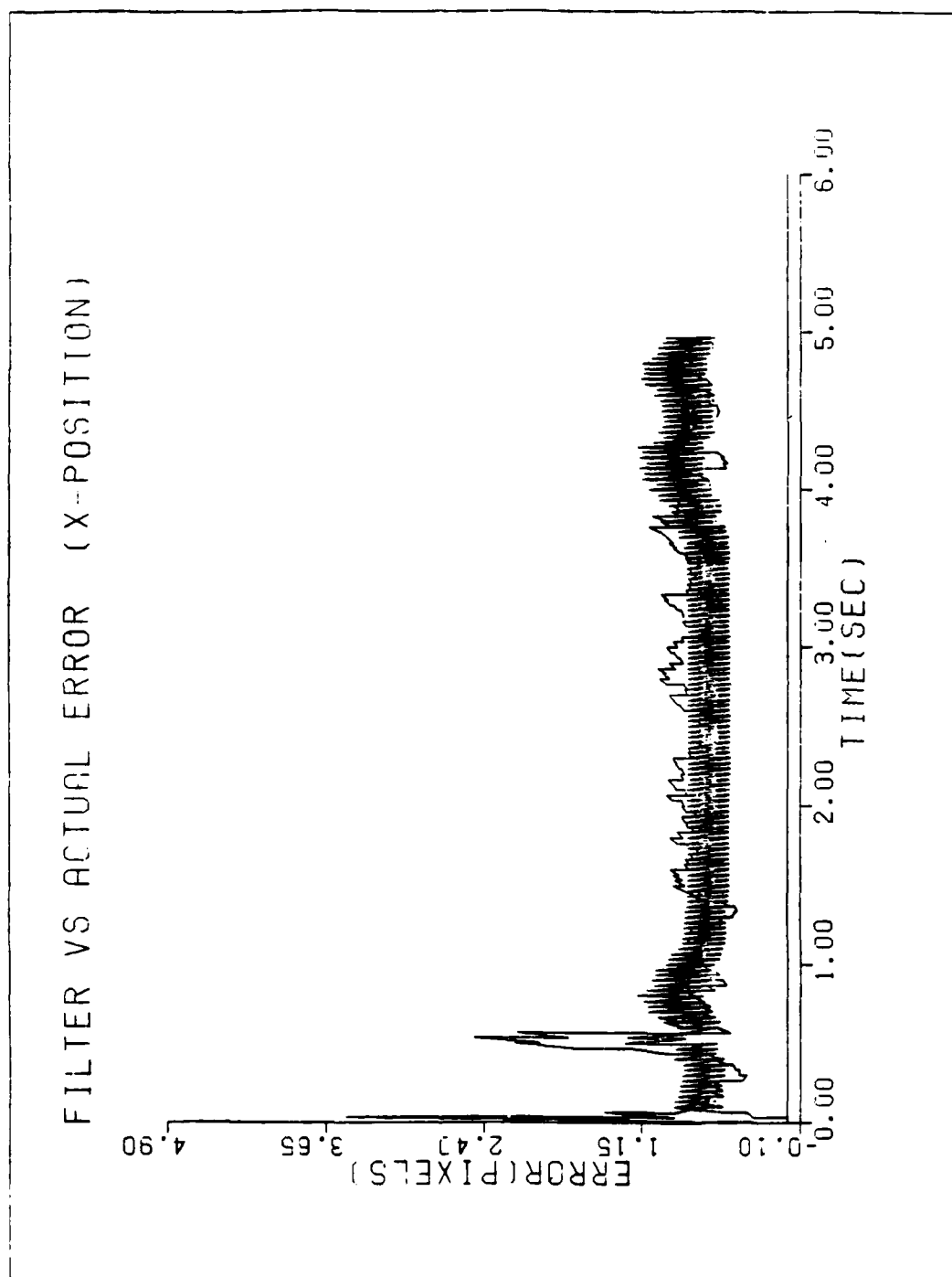


Figure D19 Performance plots for T1207

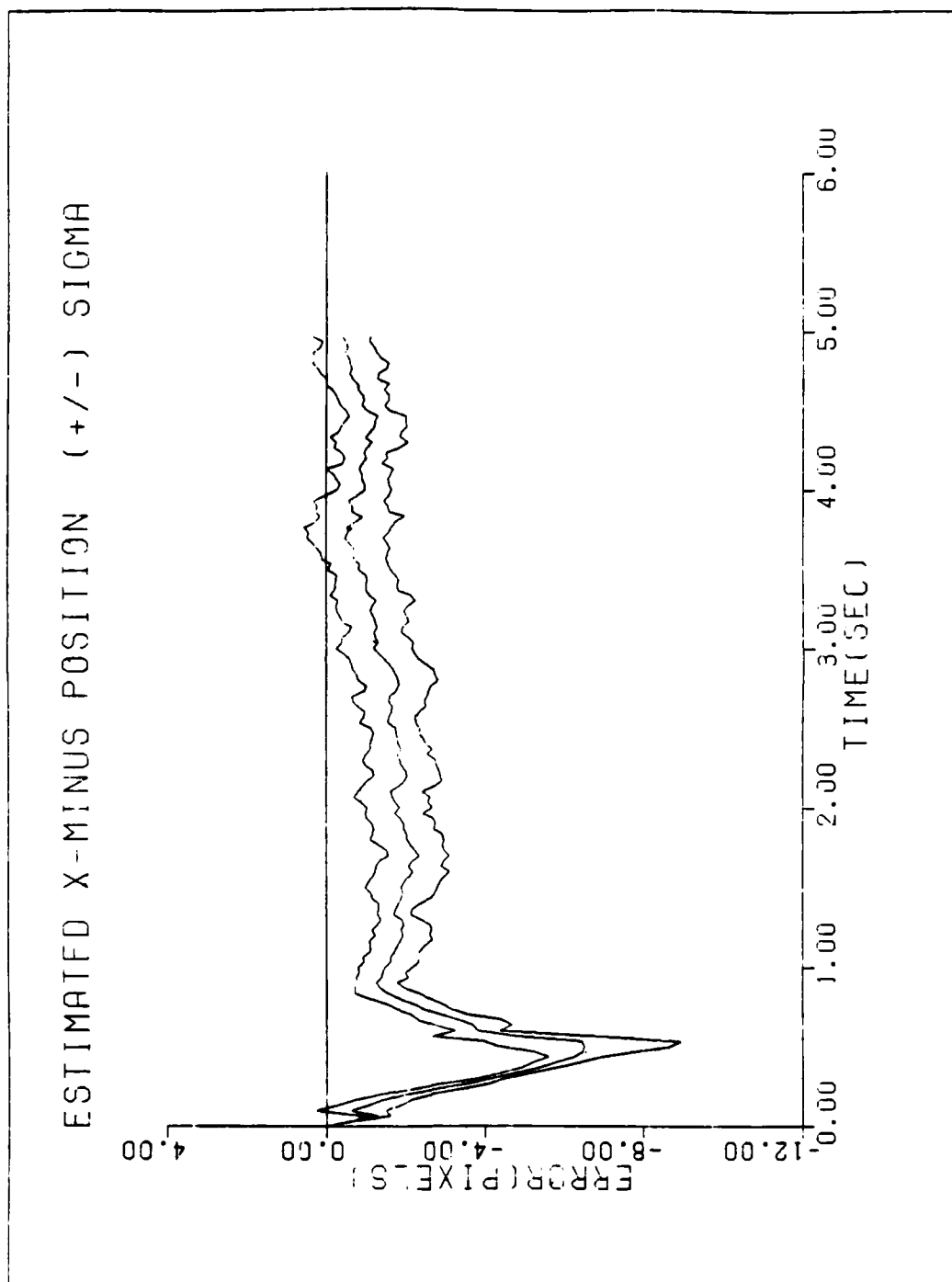


Figure D20 Performance plots for T1207

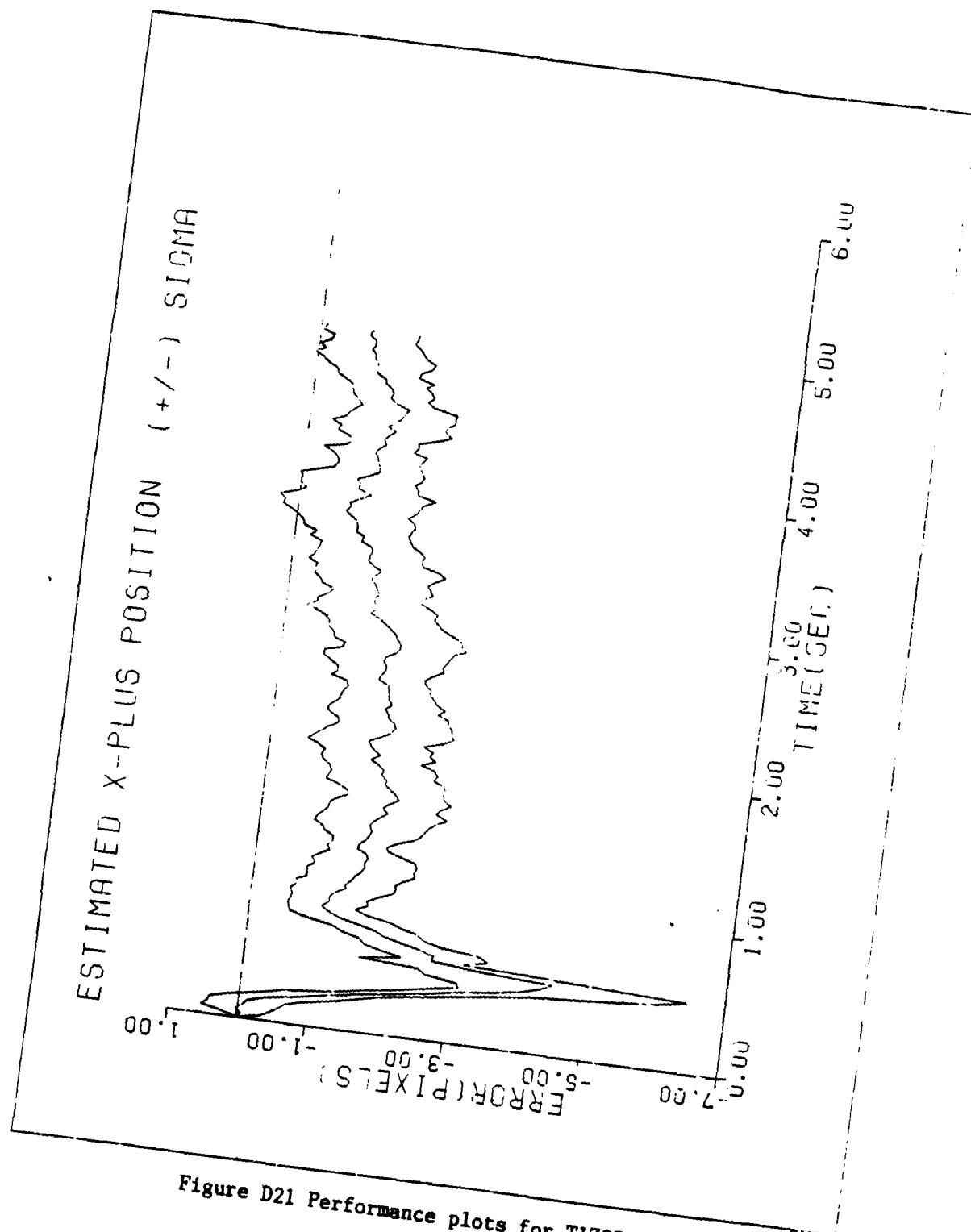


Figure D21 Performance plots for T1207

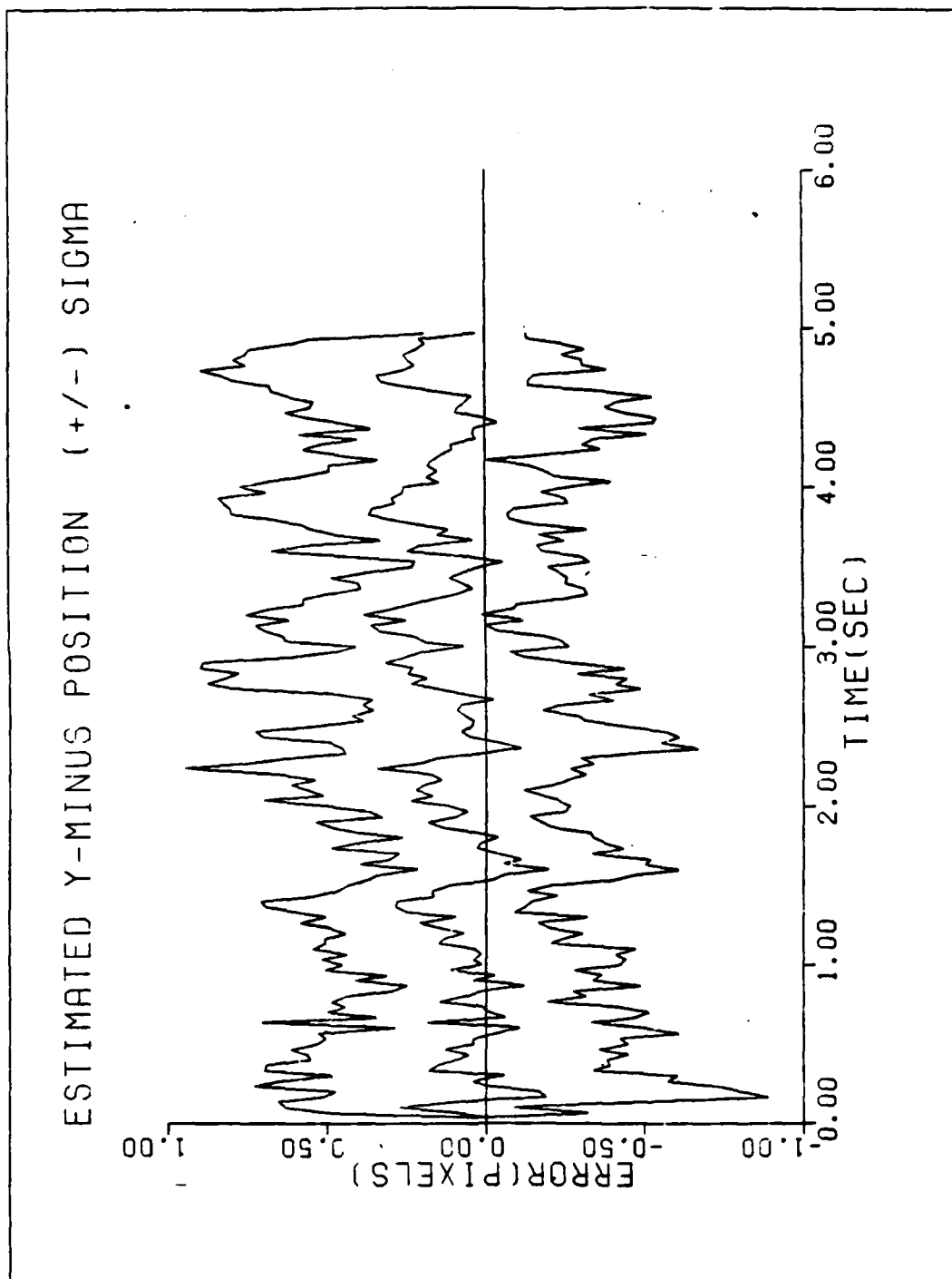


Figure D22 Performance plots for T1Z07

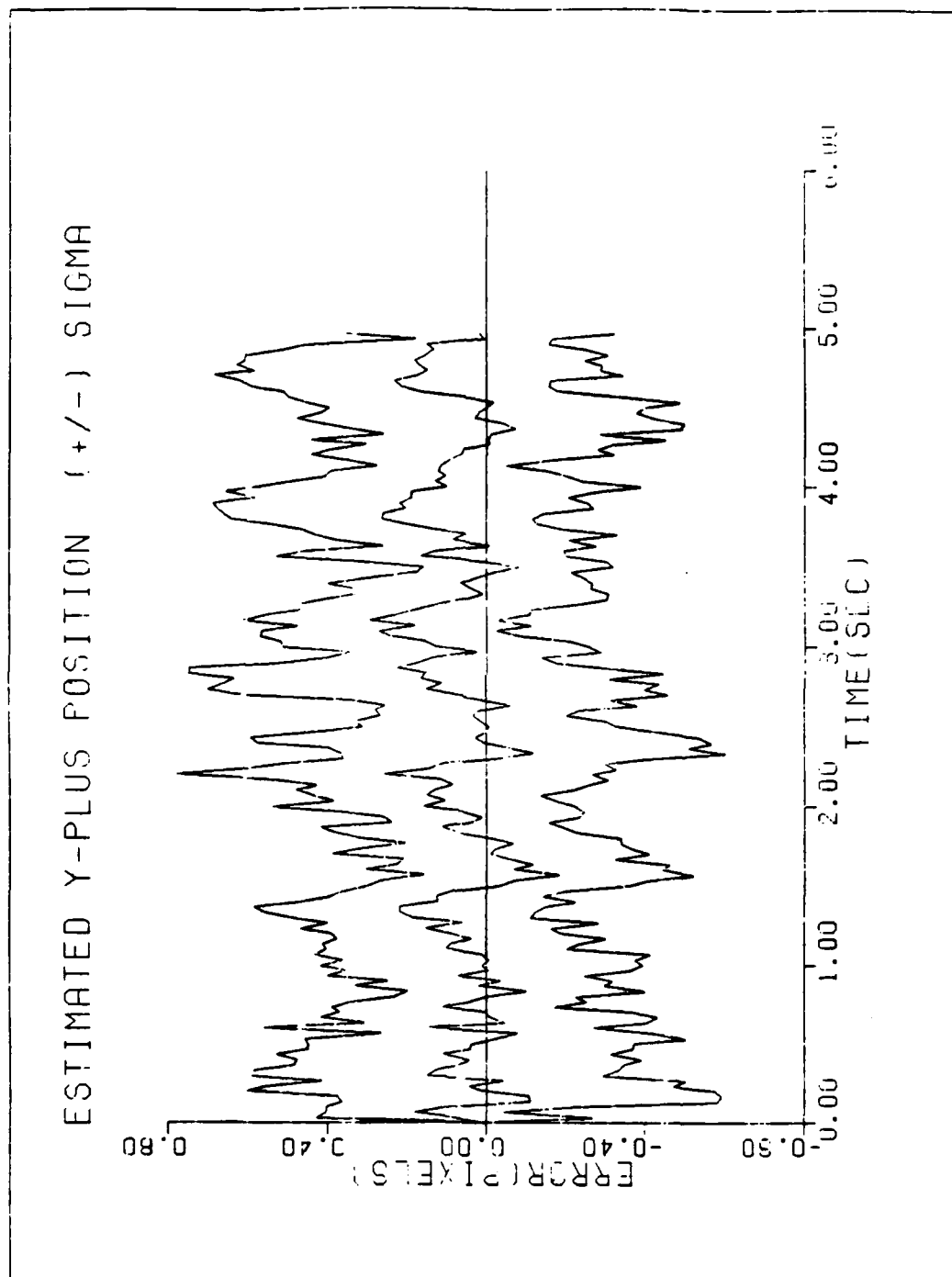


Figure D23 Performance plots for T1Z07

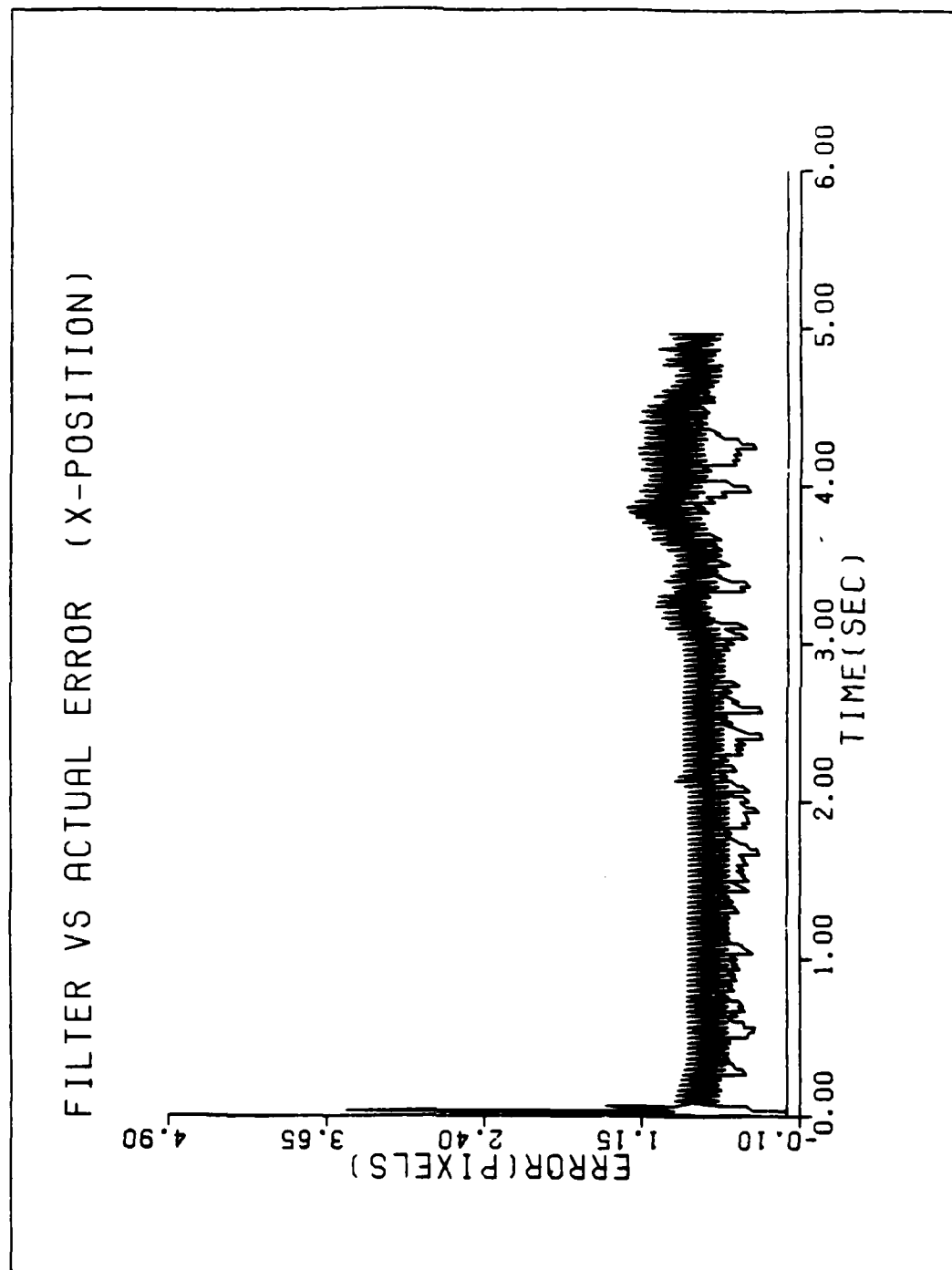


Figure D24 Performance plots T2G10Z07

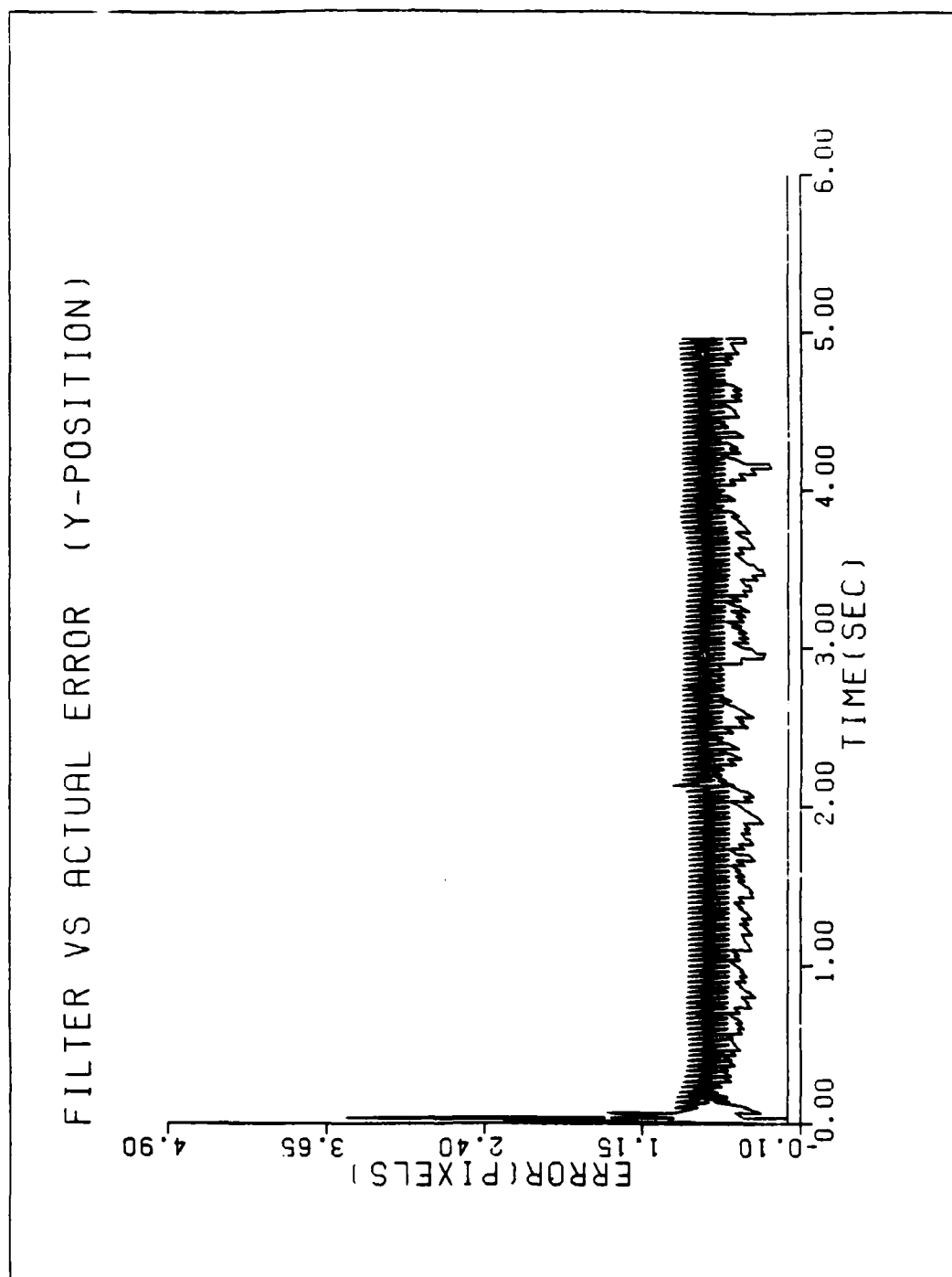


Figure D25 Performance plots for T2G10Z07

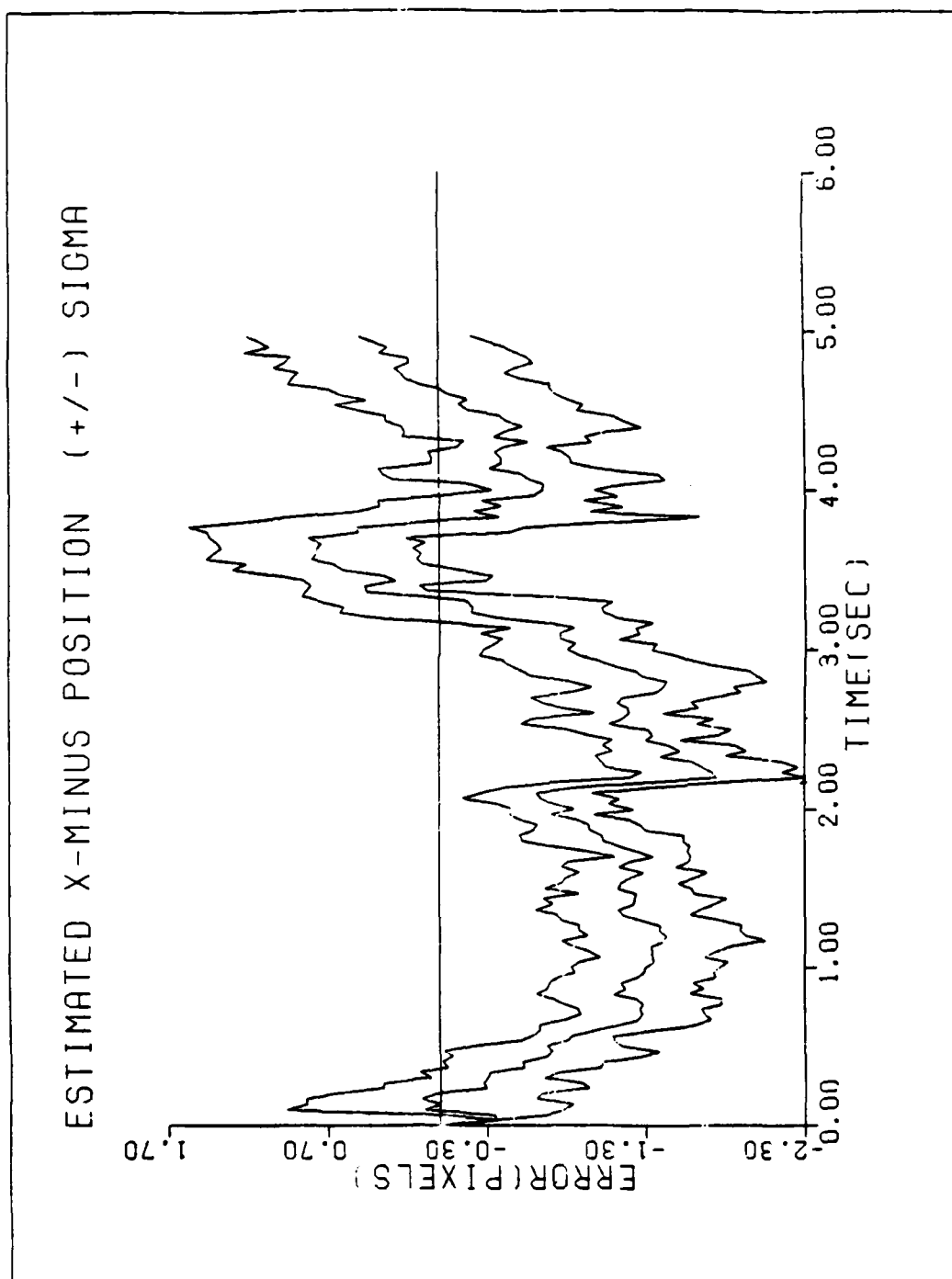


Figure D26 Performance plots for T2G10Z07

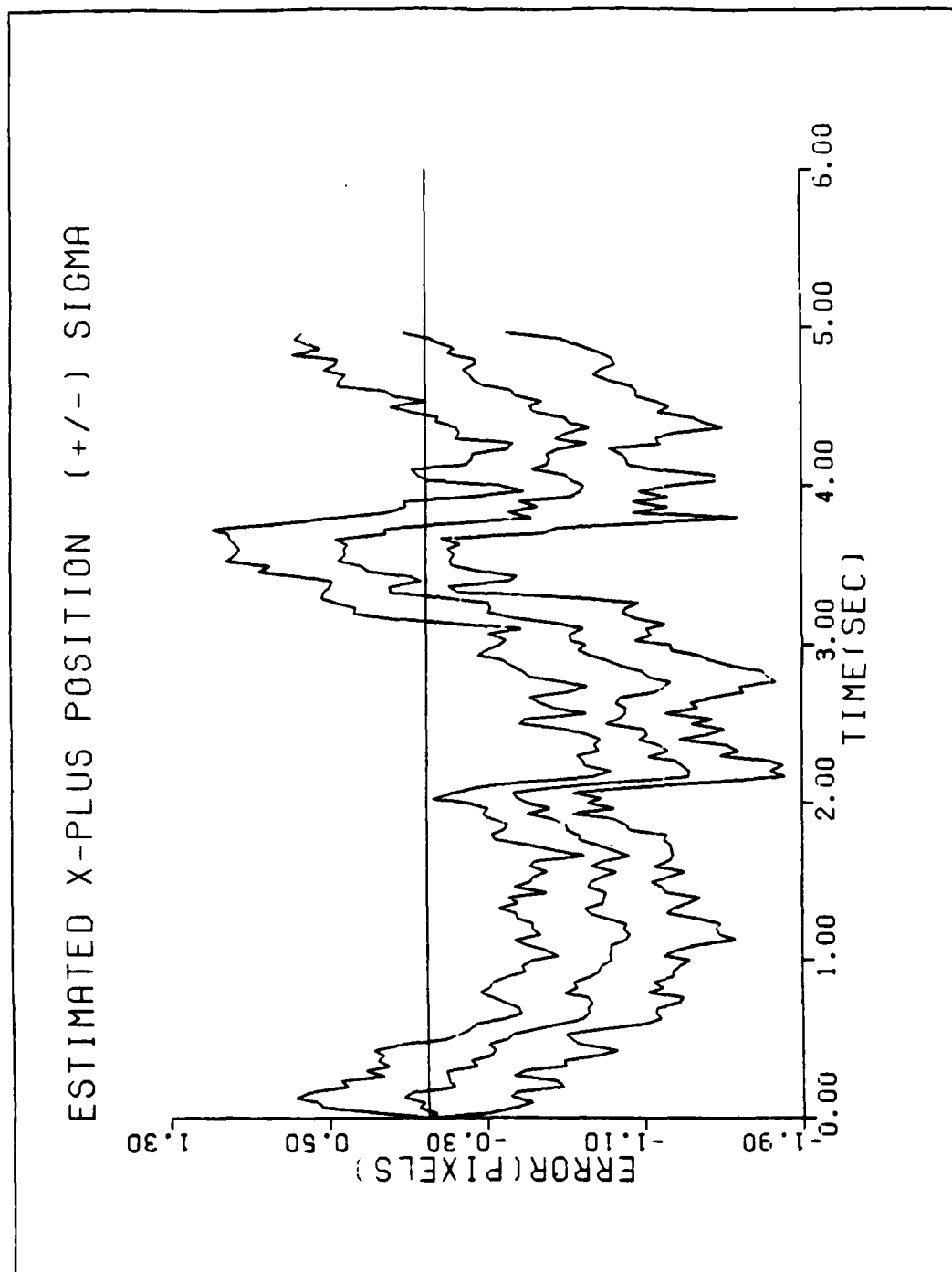


Figure D27 Performance plots for T2G10Z07

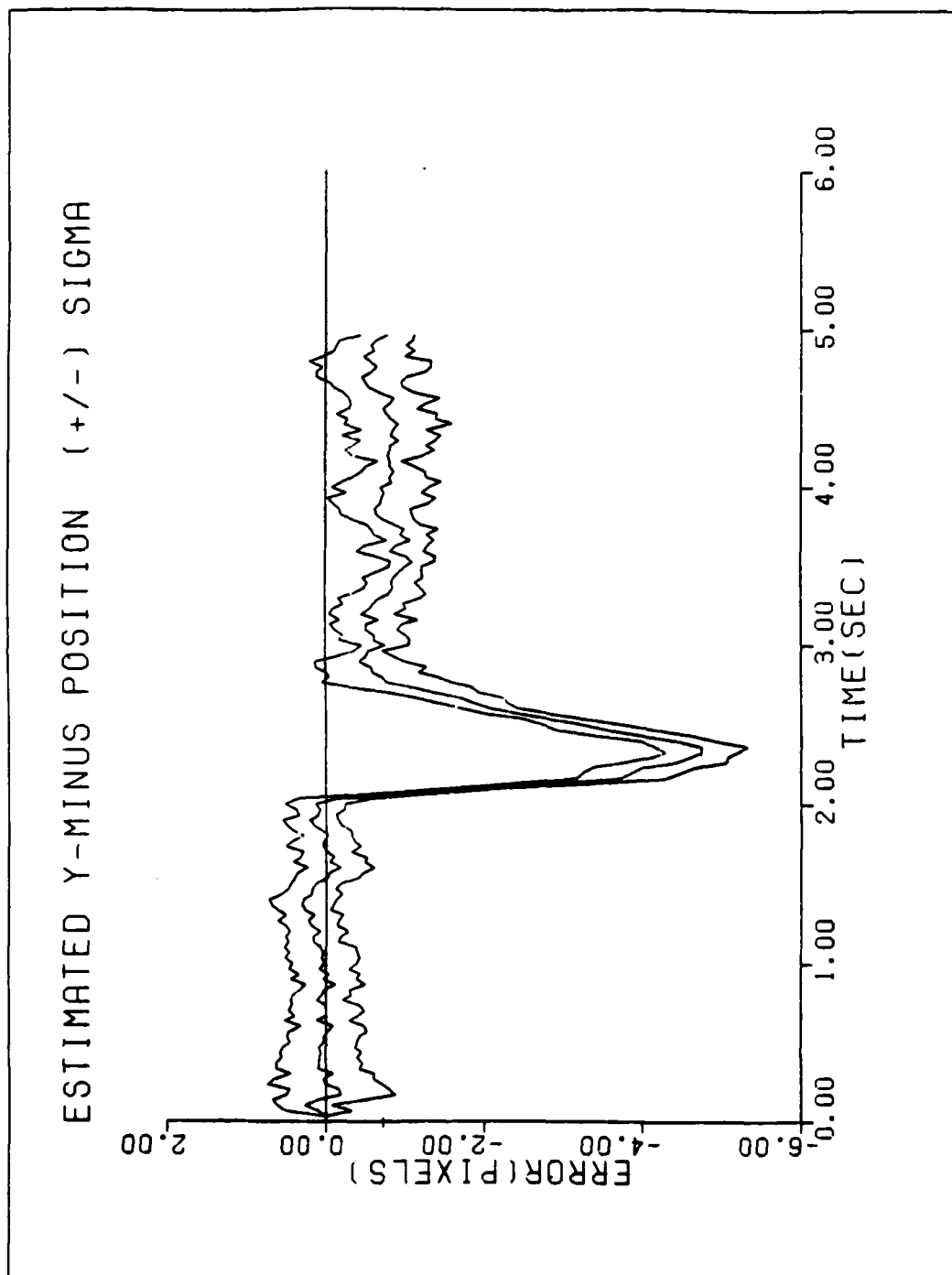


Figure D28 Performance plots for T2G10Z07

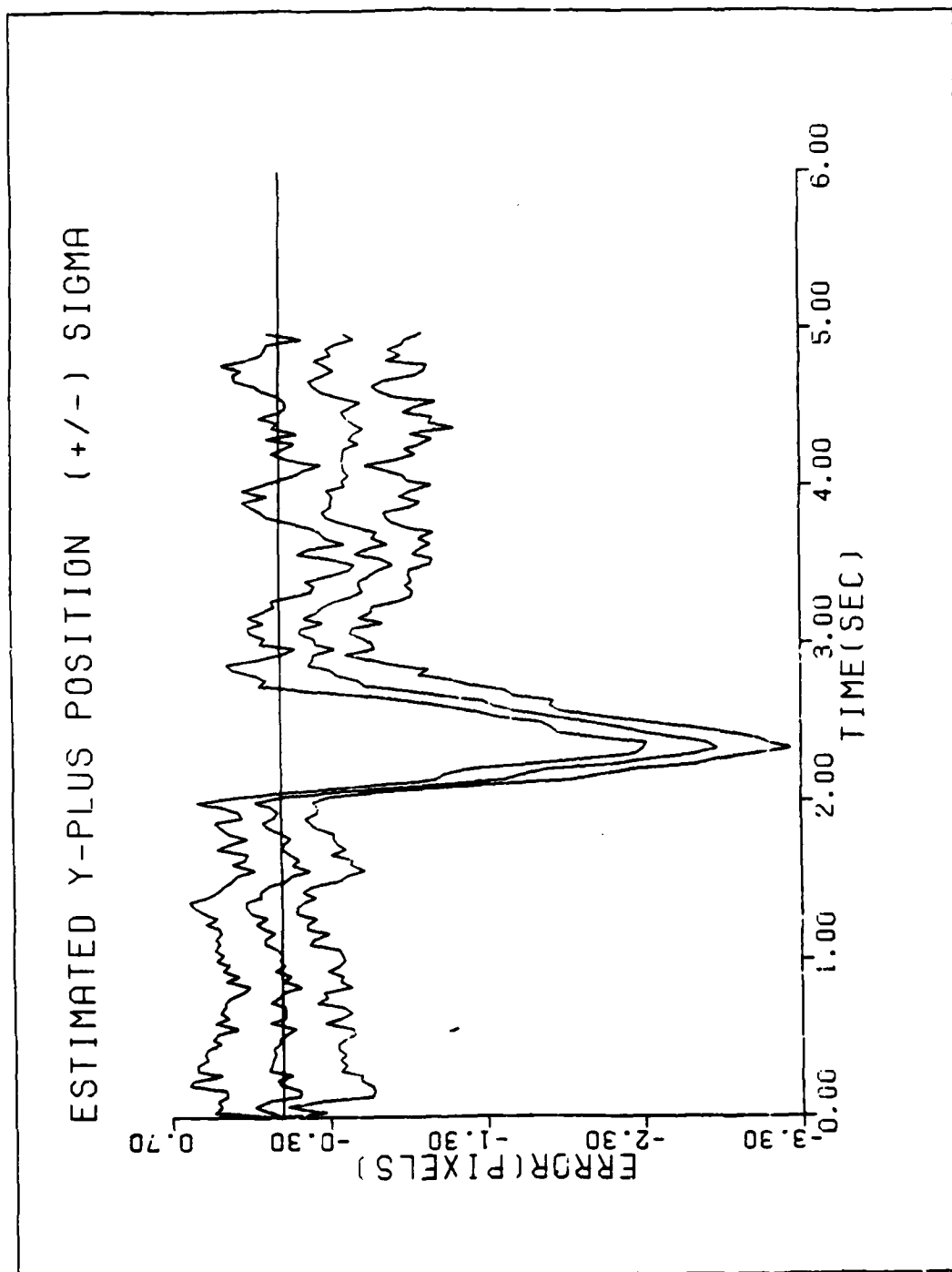


Figure D29 Performance plots for T2G10Z07

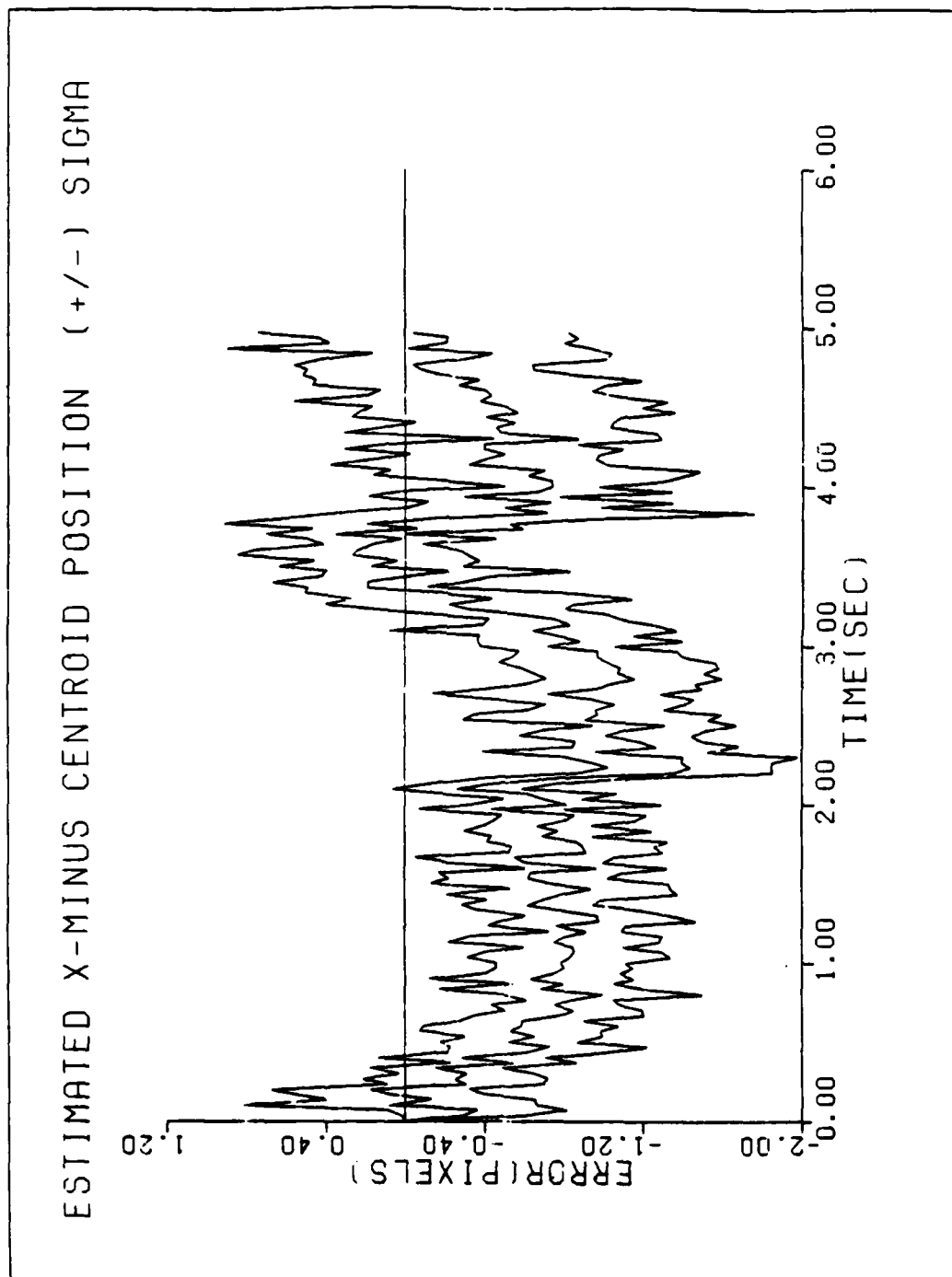


Figure D30 Performance plots for T2G10Z07

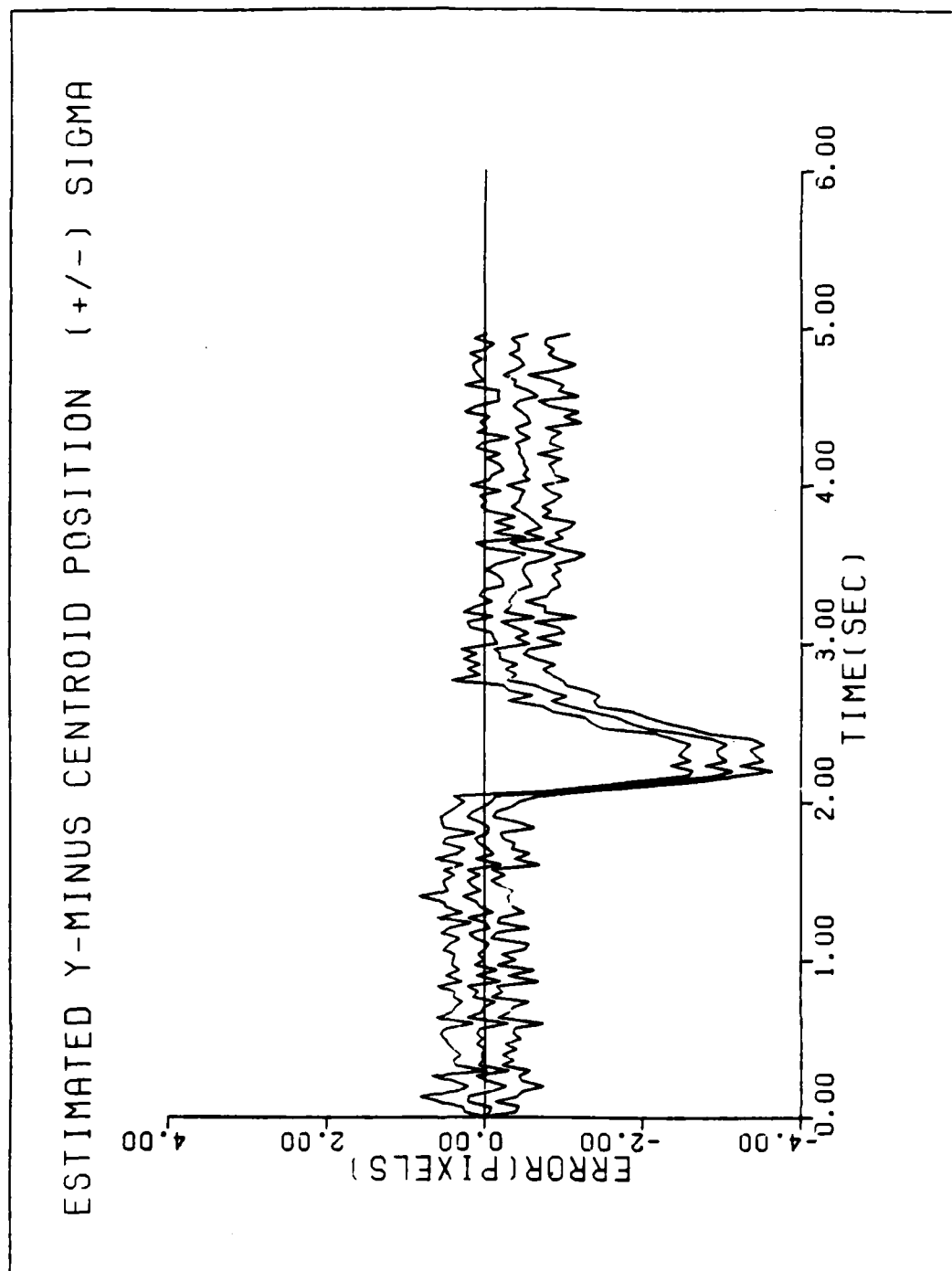


Figure D31 Performance plots for T2G10Z07

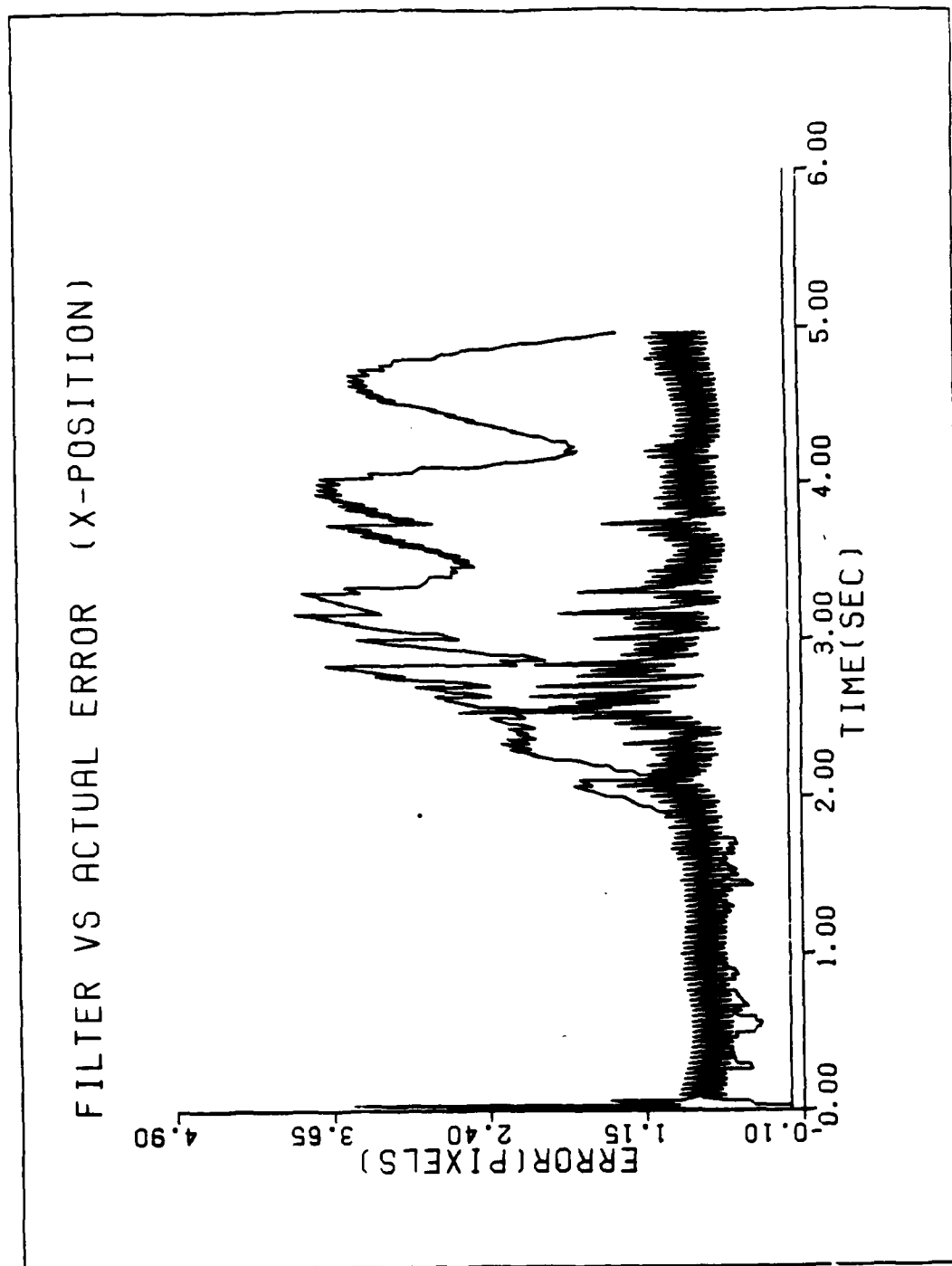


Figure D32 Performance plots for T1Z05

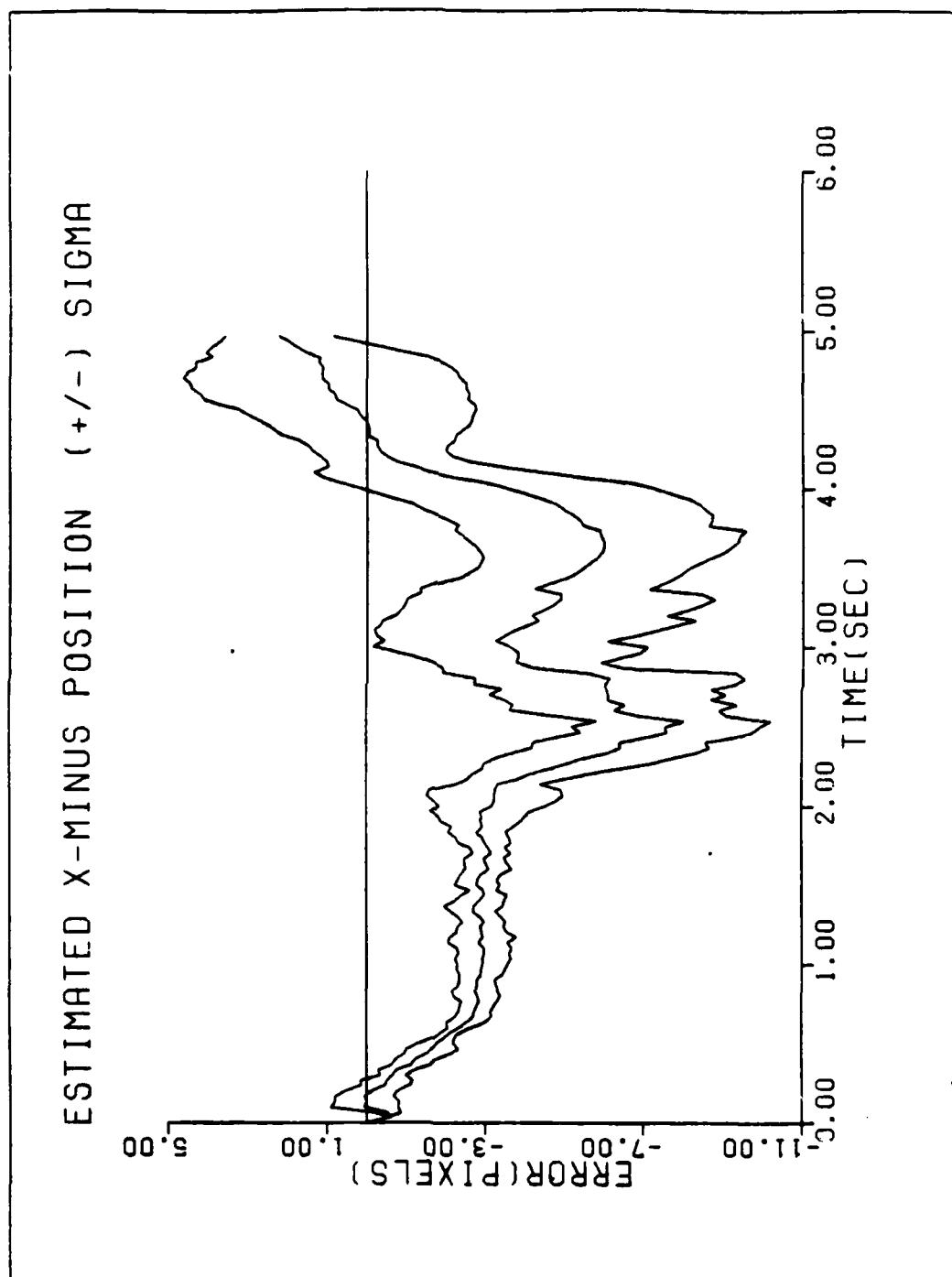


Figure D33 Performance plots for T1Z05

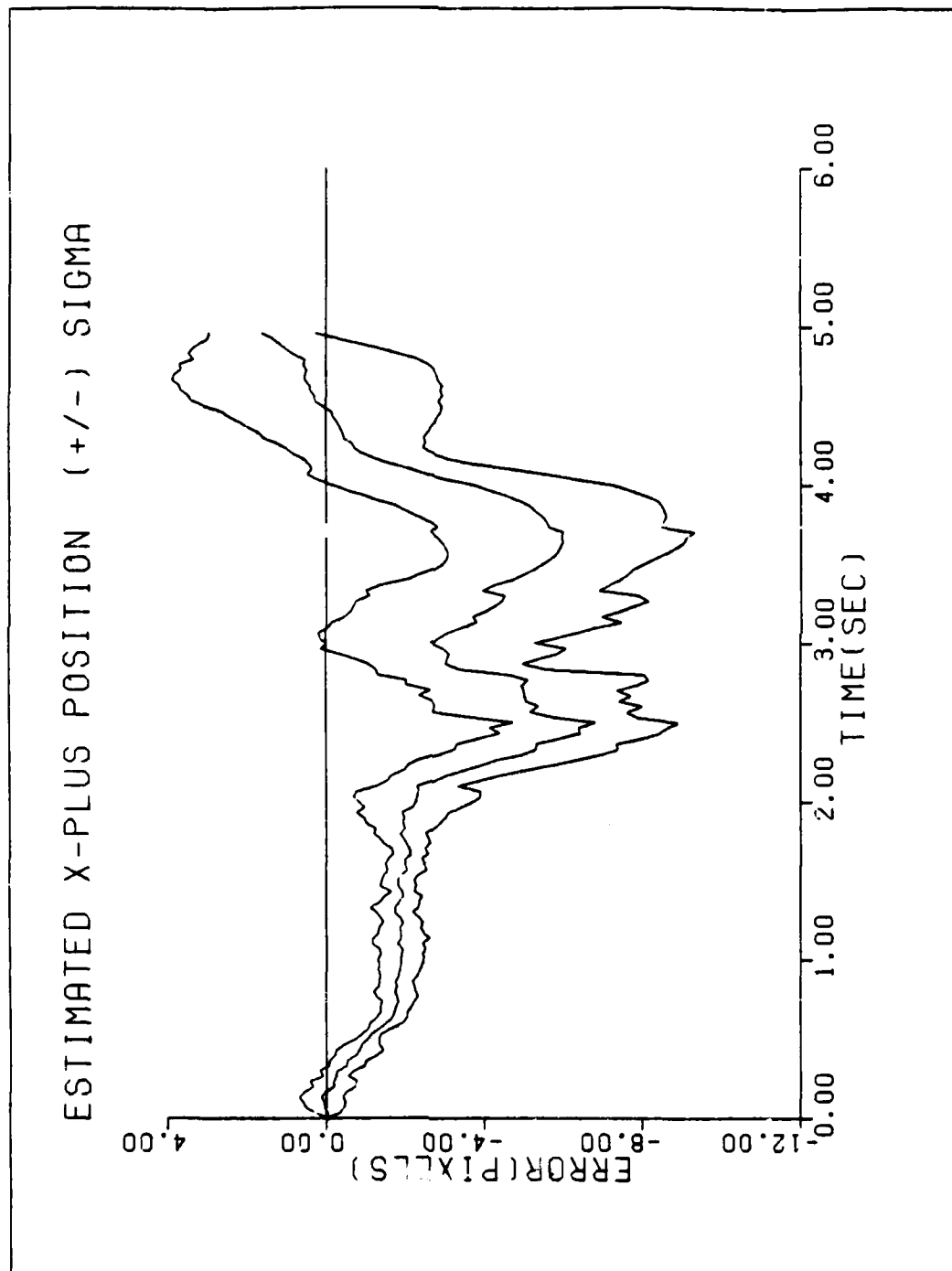


Figure D34 Performance plots for T1205

AD-A163 838

CHARACTERISTICS OF BAYESIAN MULTIPLE MODEL ADAPTIVE
ESTIMATION FOR TRACKING (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI... A S NETZER

4/4

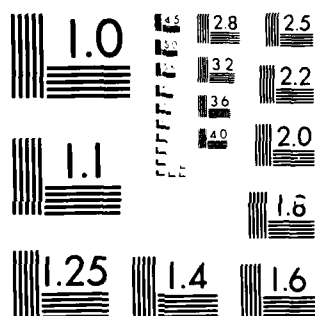
UNCLASSIFIED

DEC 85 AFIT/GAE/ENG/85D-2

F/G 17/5

ML

								END					
								FILED					
								UPL					



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

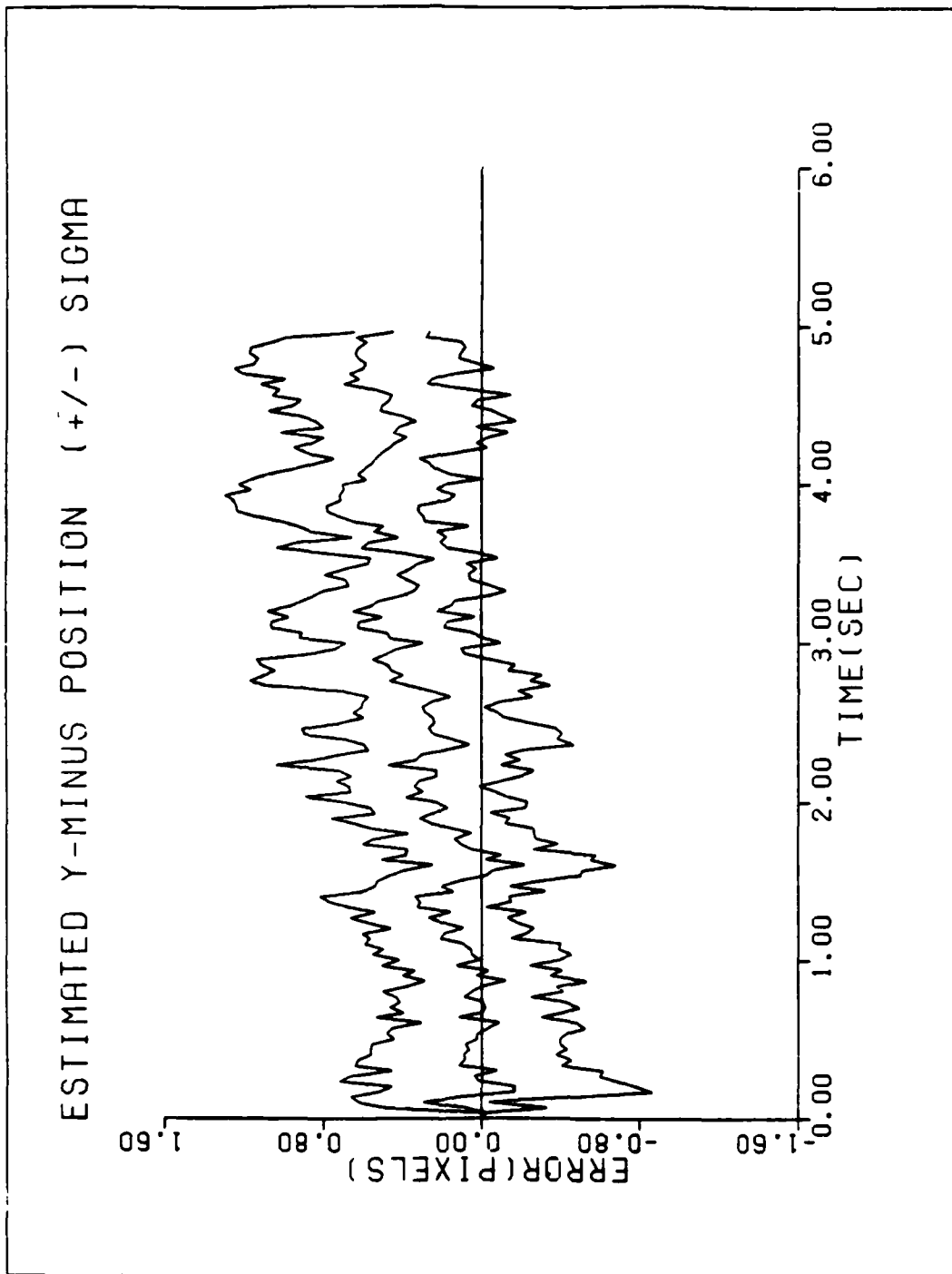


Figure D35 Performance plots for T1Z05

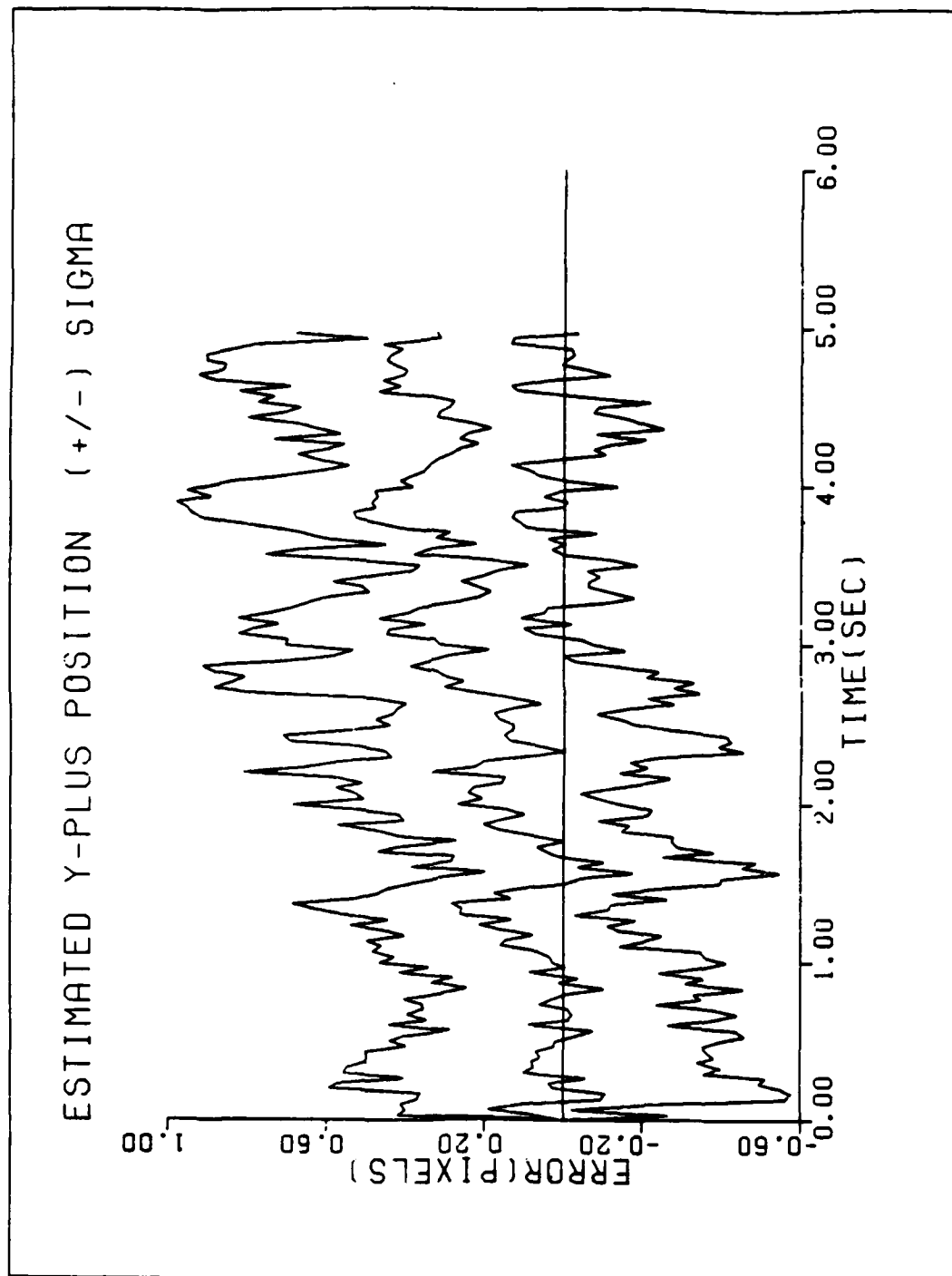


Figure D36 Performance plots for T1205

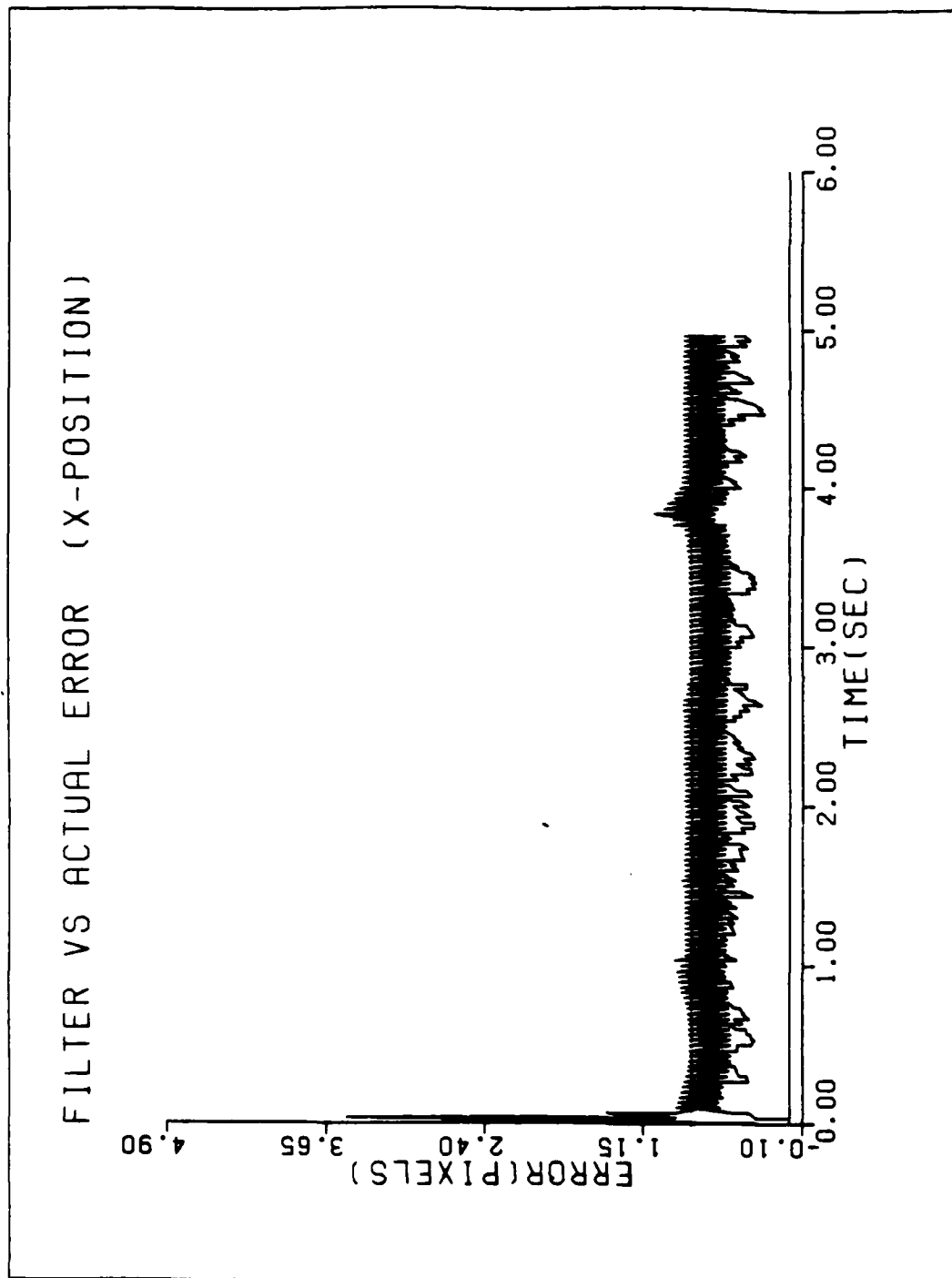


Figure D37 Performance plots for T1Z05NI

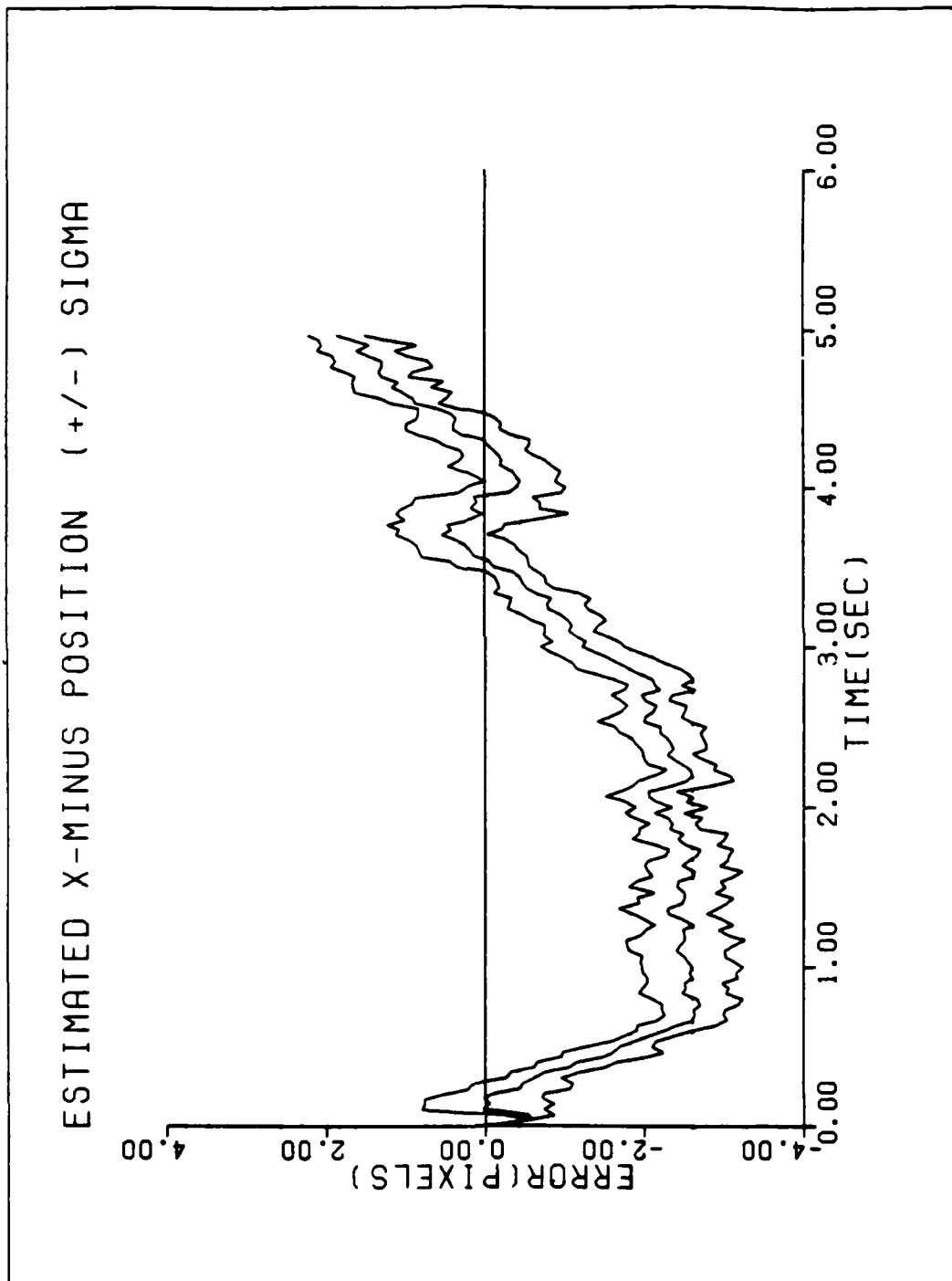


Figure D38 Performance plots for T1Z05NI

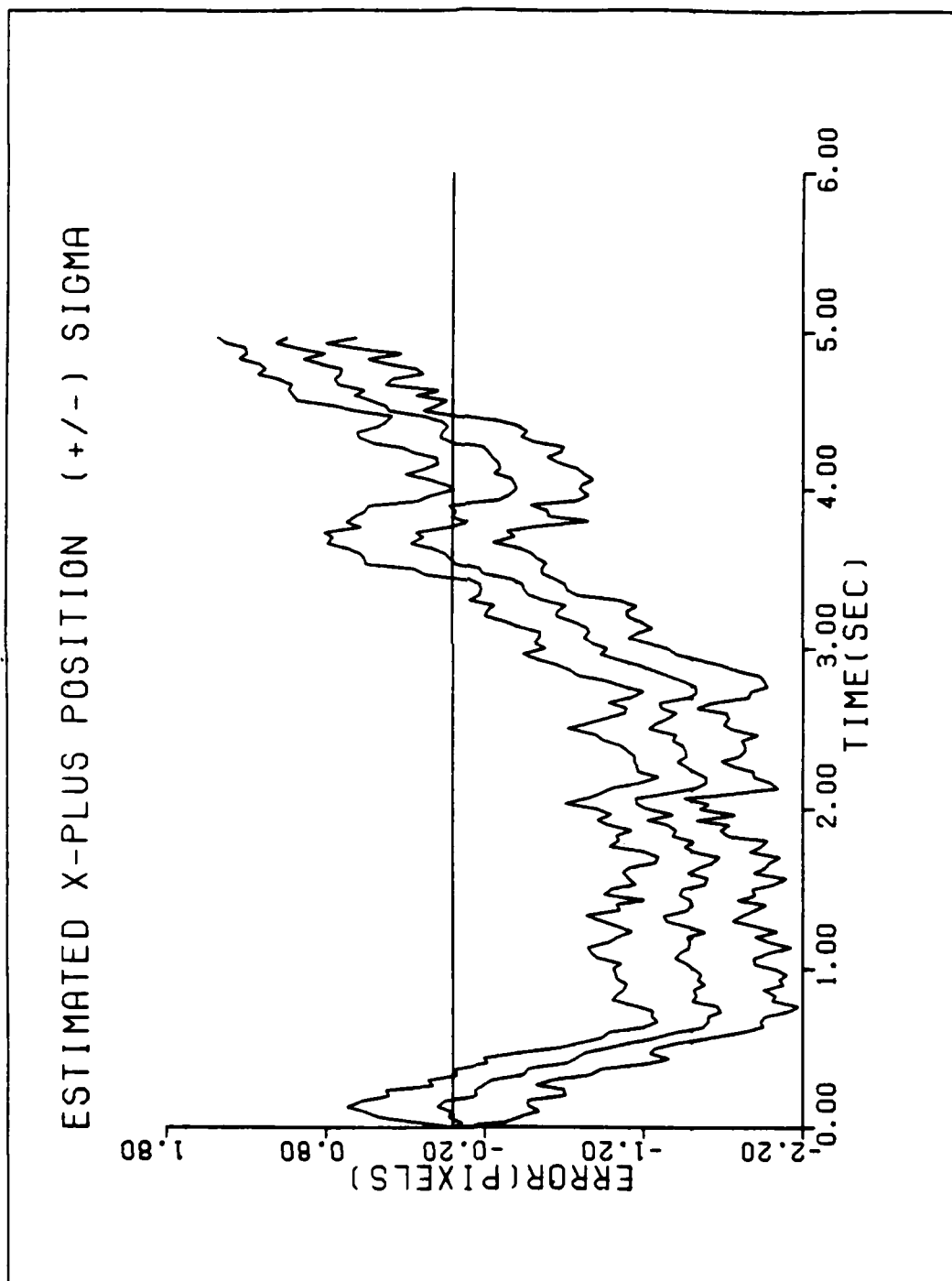


Figure D39 Performance plots for T1Z05NI

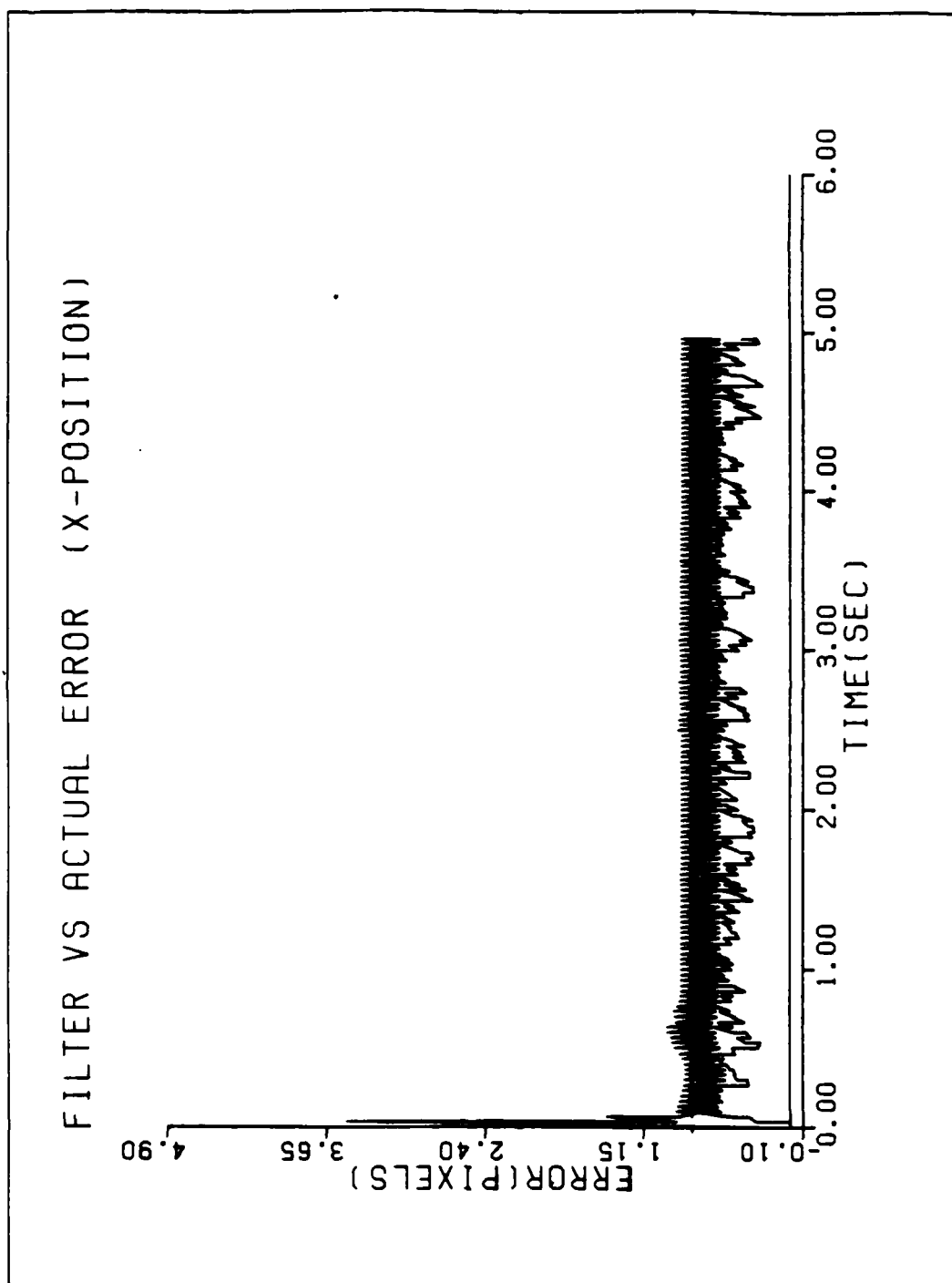


Figure D40 Performance plots for T1Z05NIRT

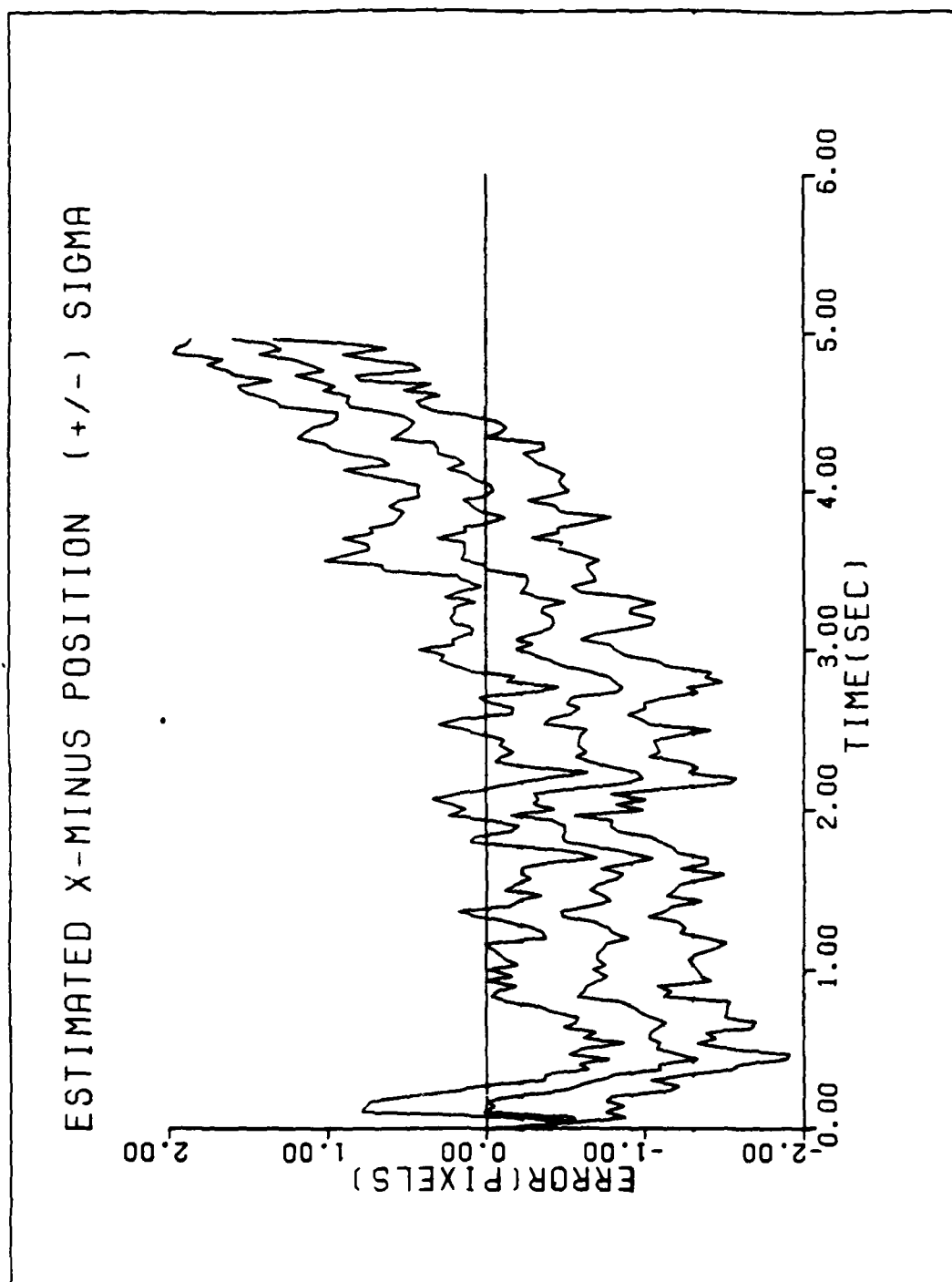


Figure D41 Performance plots for T1Z05NIRT

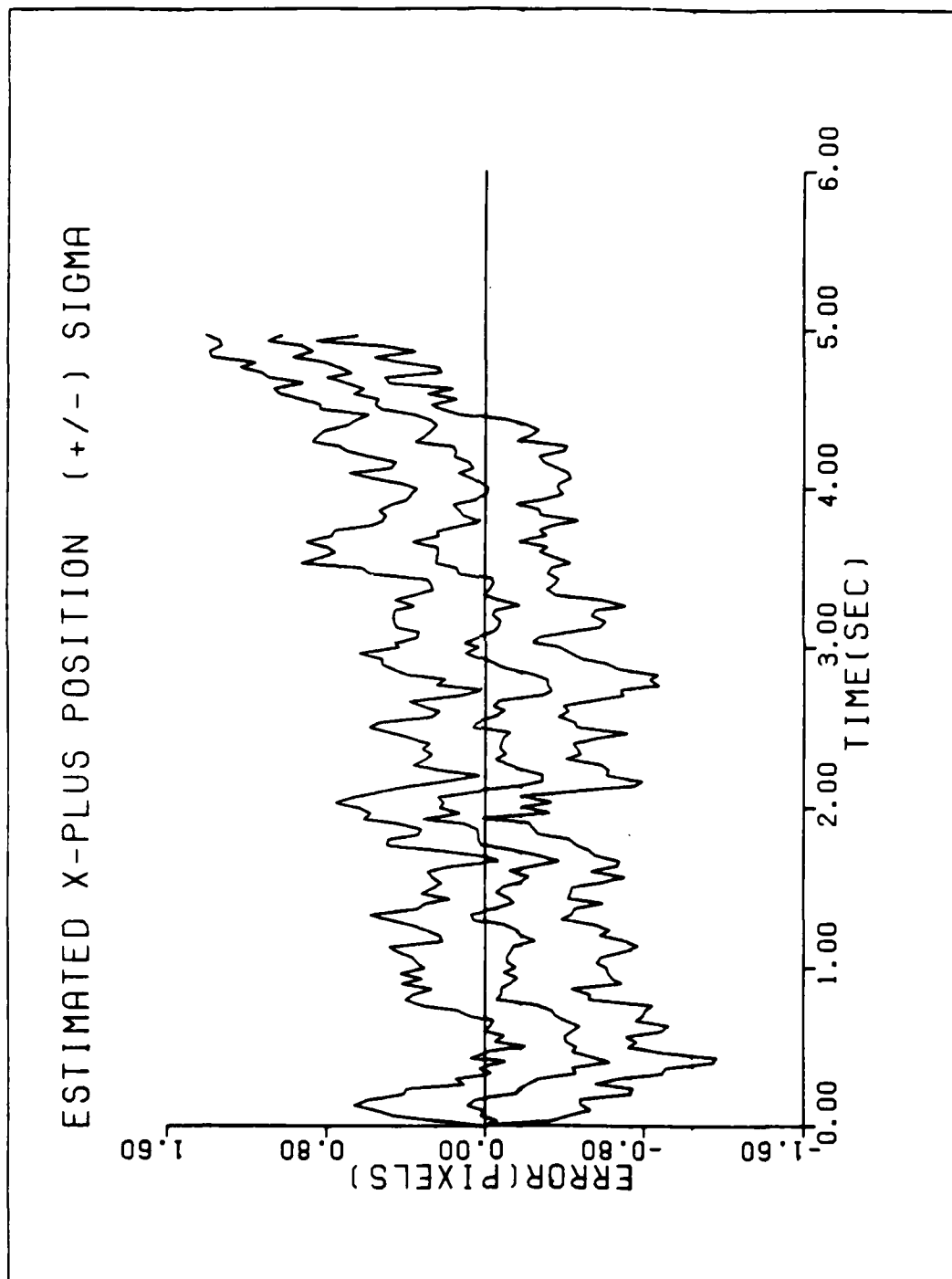


Figure D42 Performance plots for T1Z05NIRT

APPENDIX E
TARGET SHAPE SENSITIVITY ANALYSIS

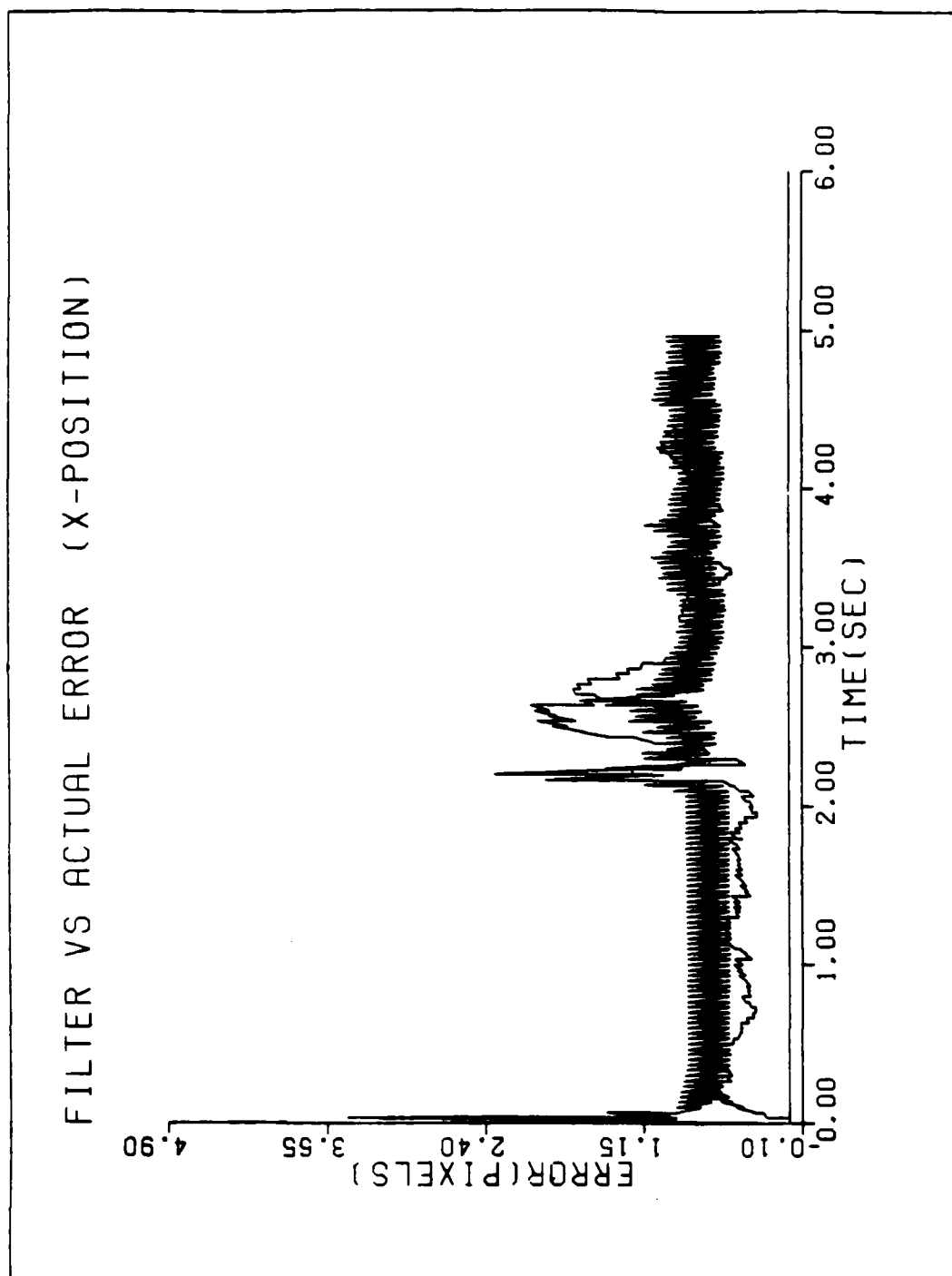


Figure E1 Performance plot for T2G10AR5SA

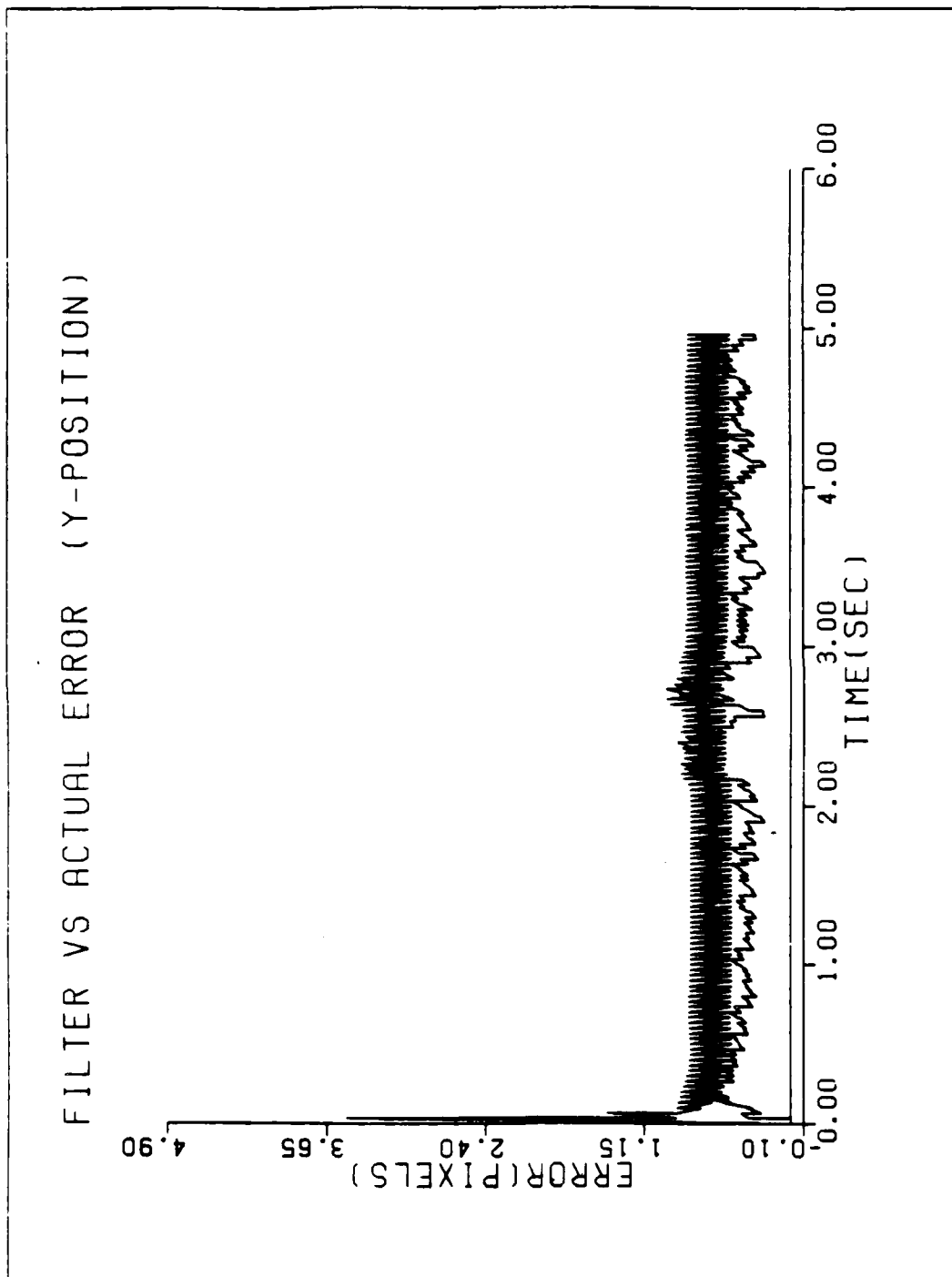


Figure E2 Performance plot for T2G10AR5SA

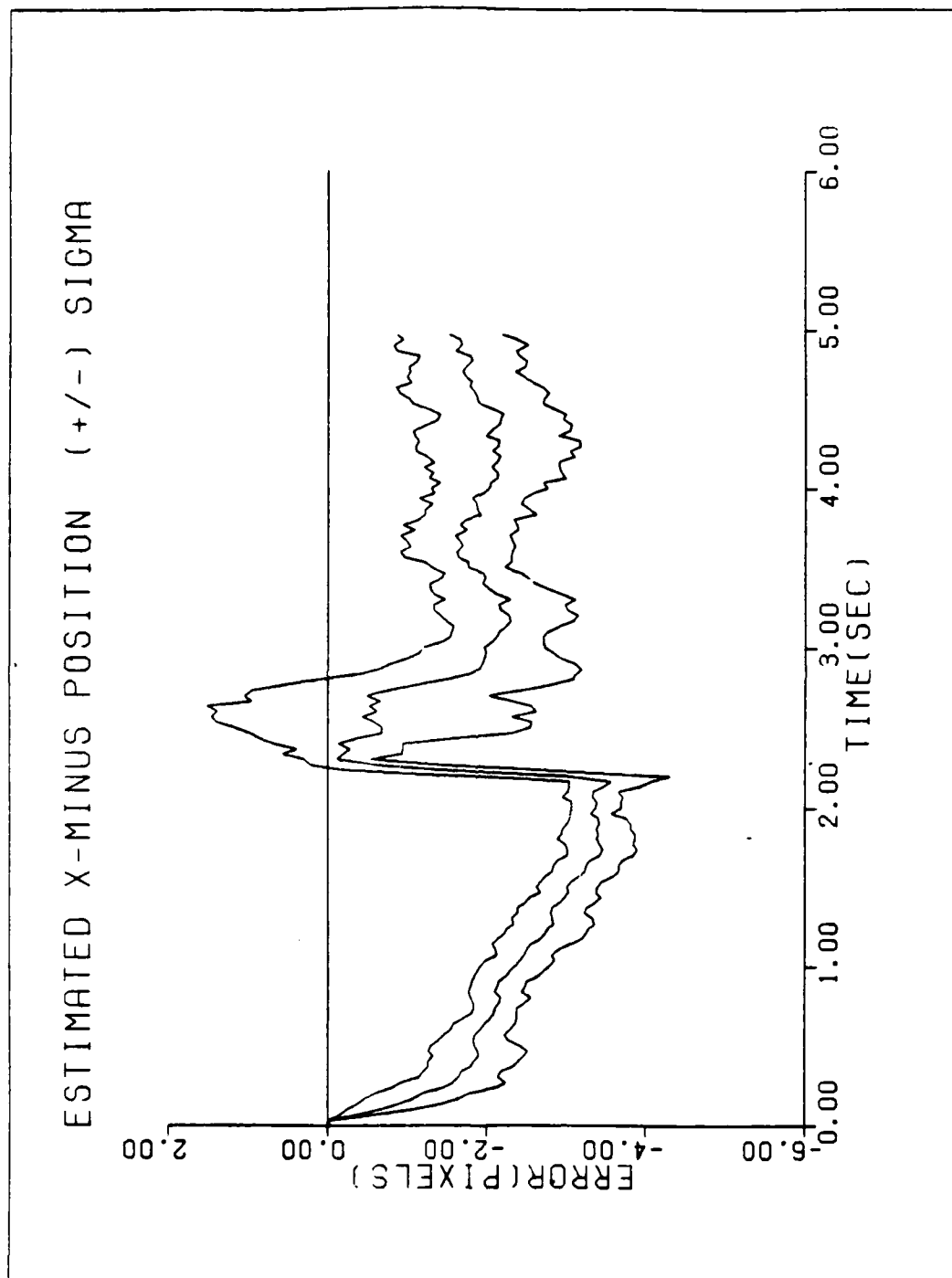


Figure E3 Performance plot for T2G10AR5SA

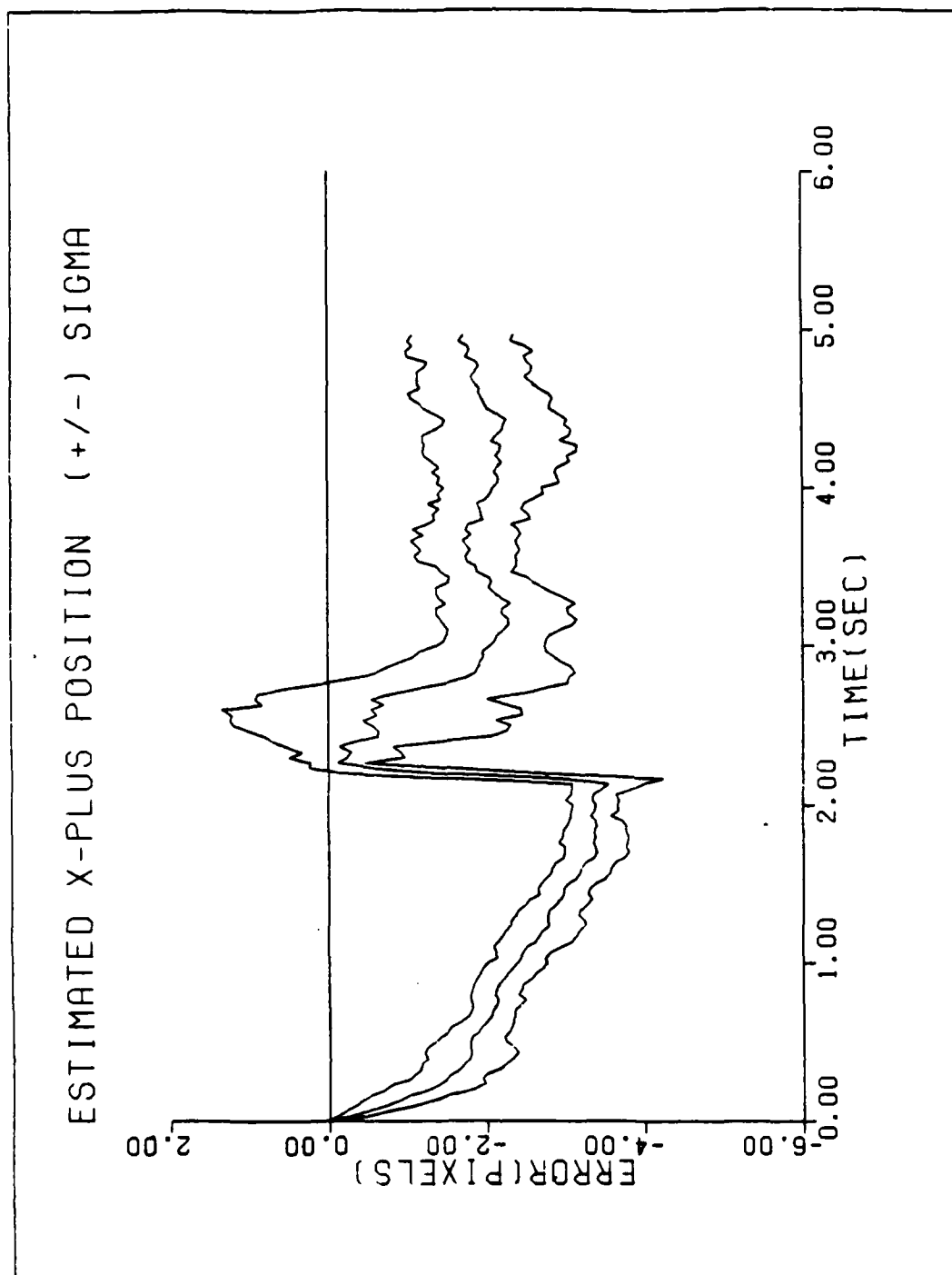


Figure E4 Performance plot for T2G10AR5SA

ESTIMATED Y-MINUS POSITION (+/-) SIGMA

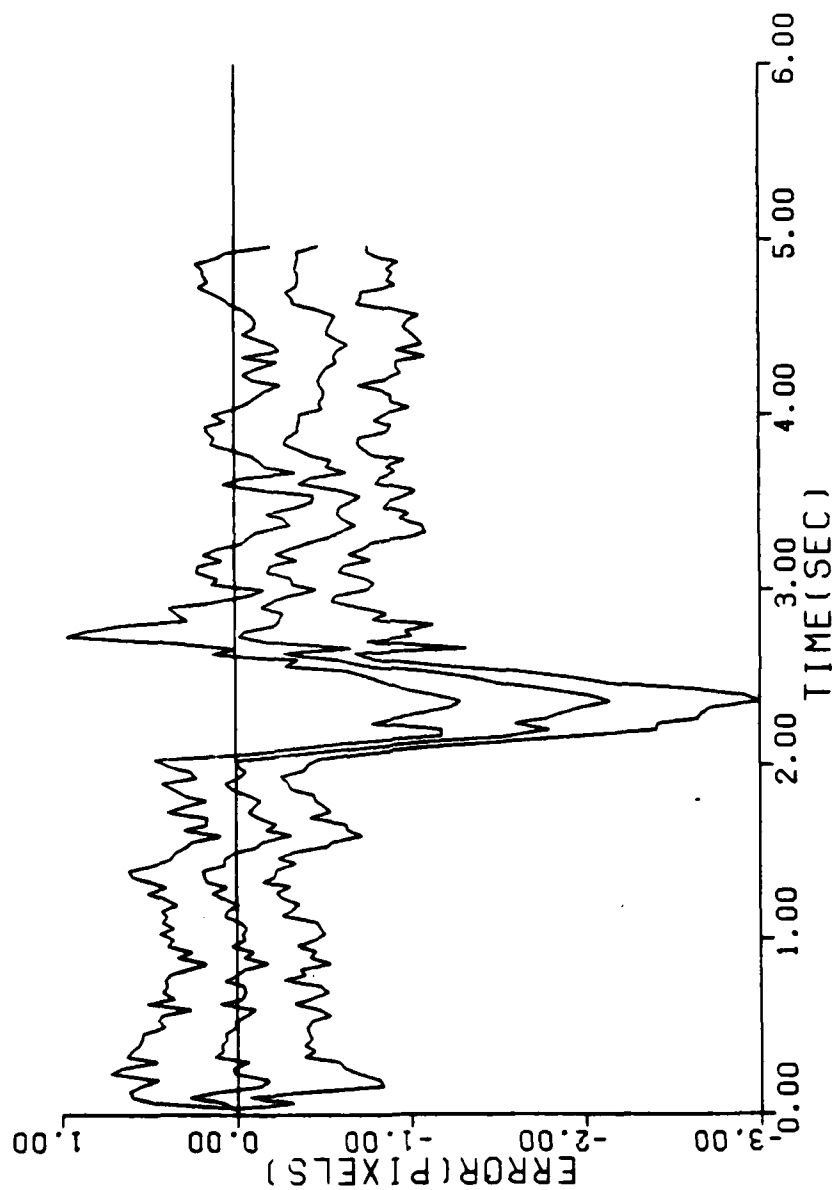


Figure E5 Performance plot for T2G10AR5SA

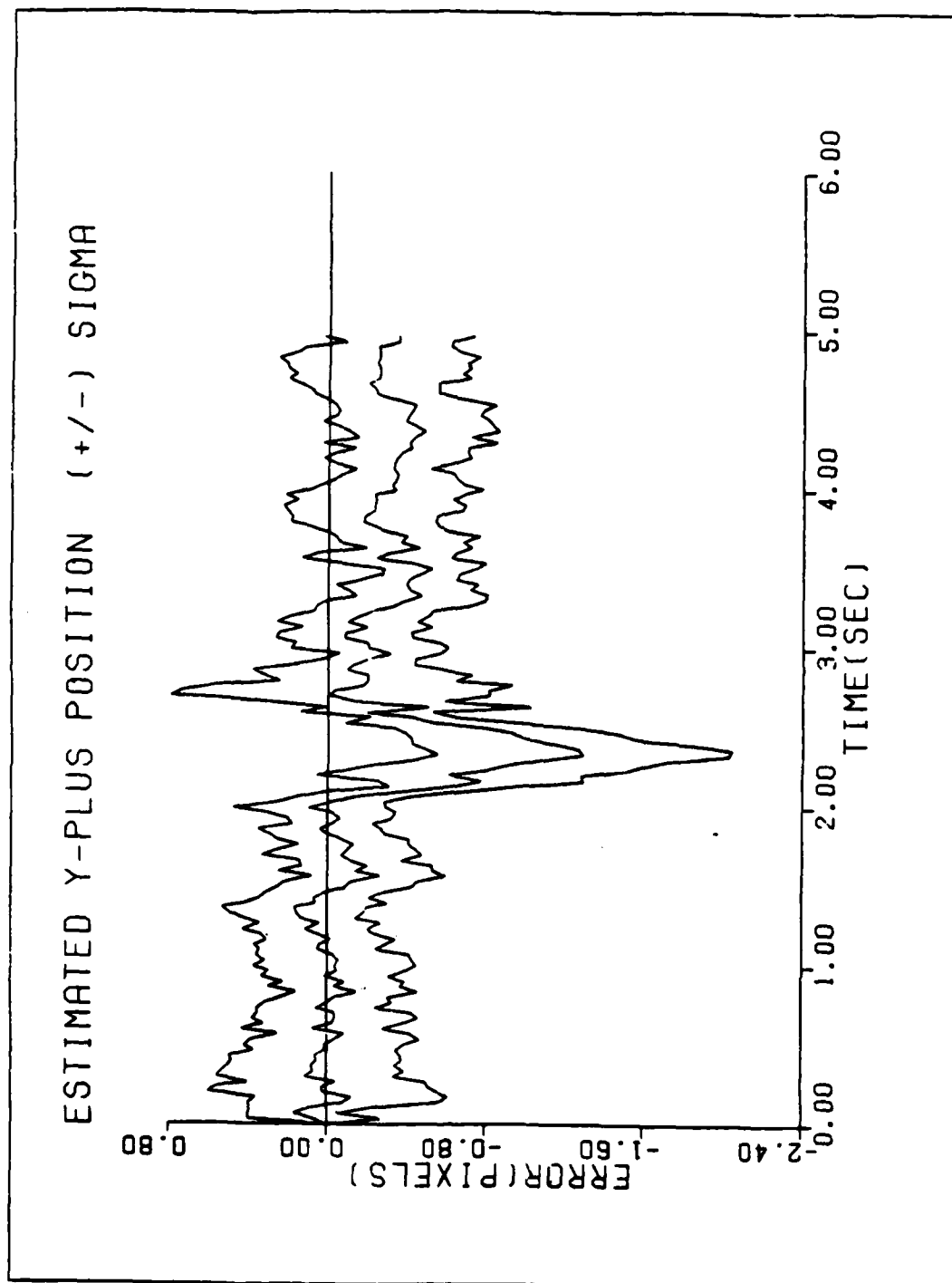


Figure E6 Performance plot for T2G10AR5SA

FILTER VS ACTUAL ERROR (X-POSITION)

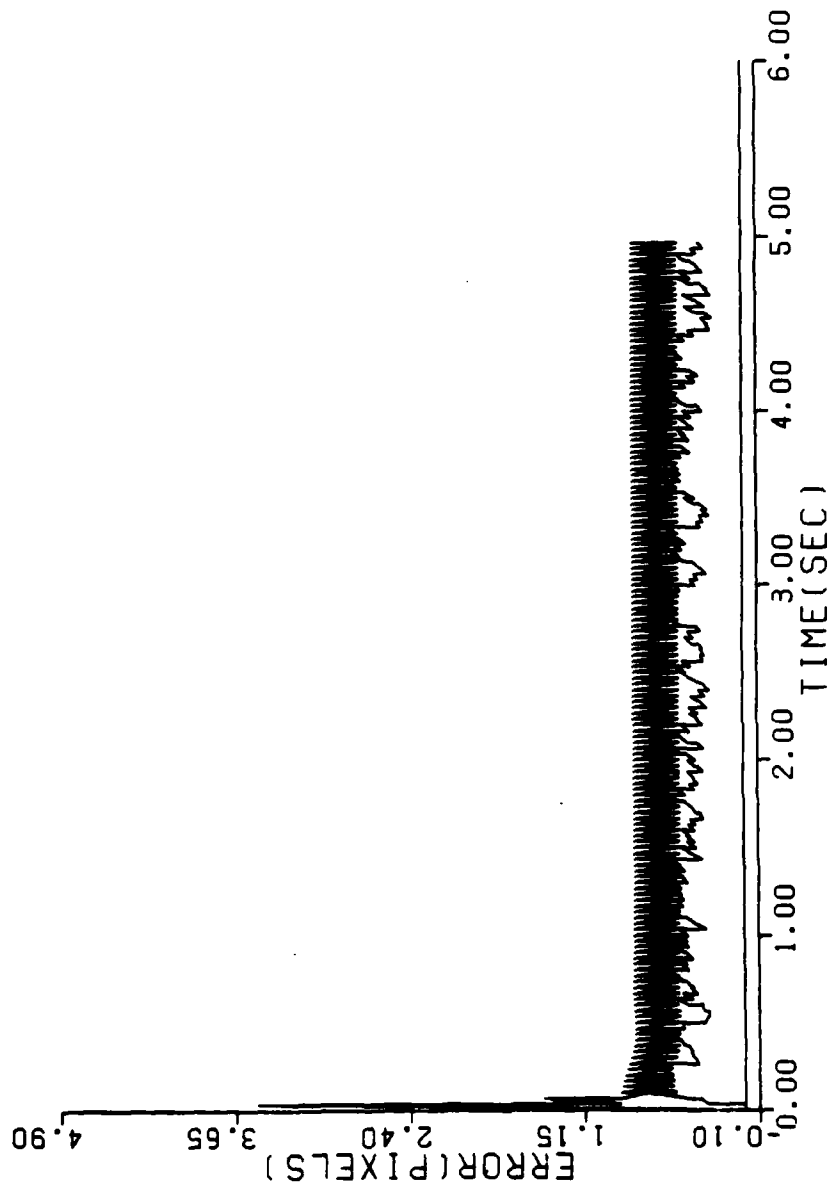


Figure E7 Performance plot for TIAR.2SA

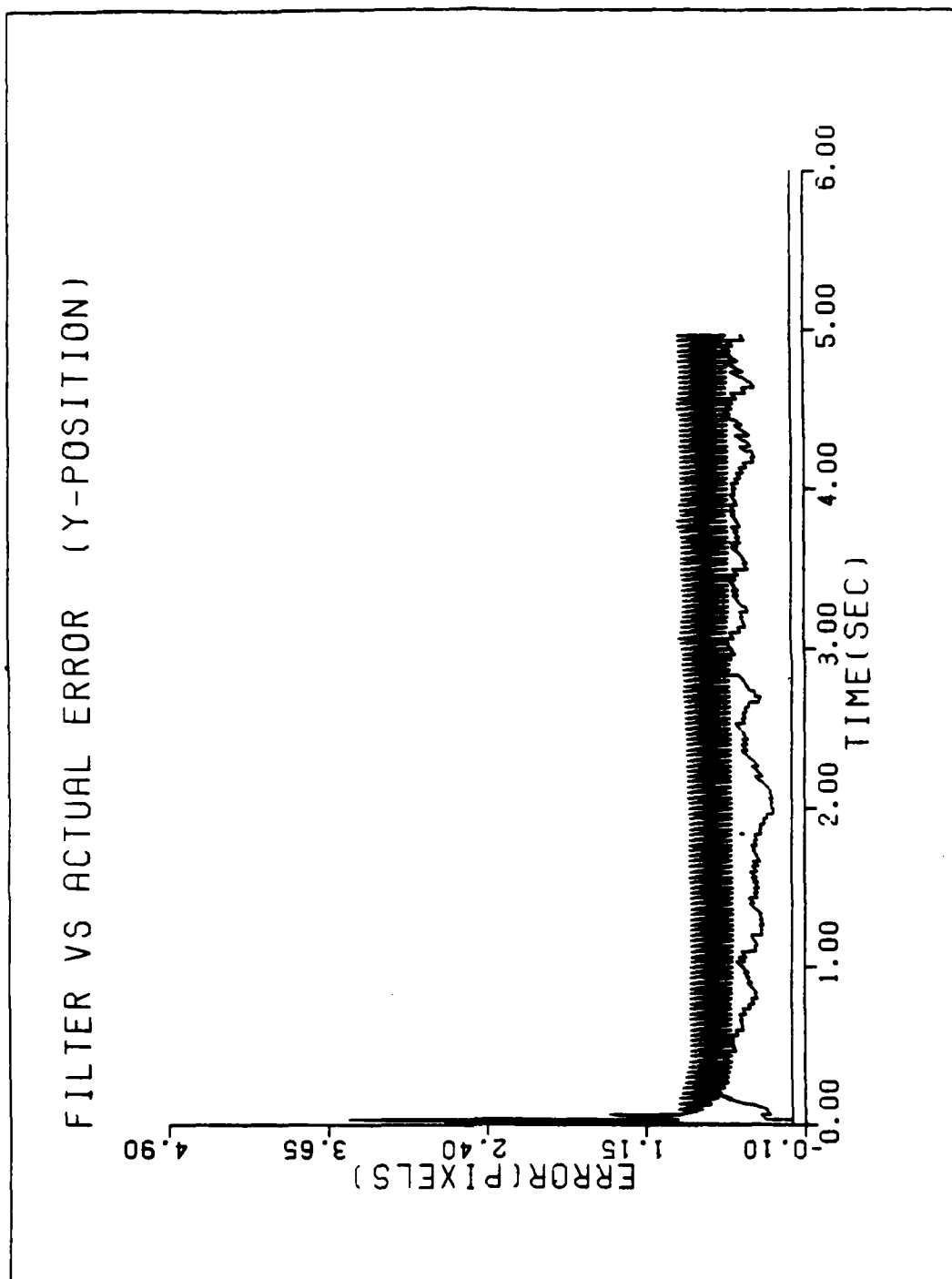


Figure E8 Performance plot for TIAR.2SA

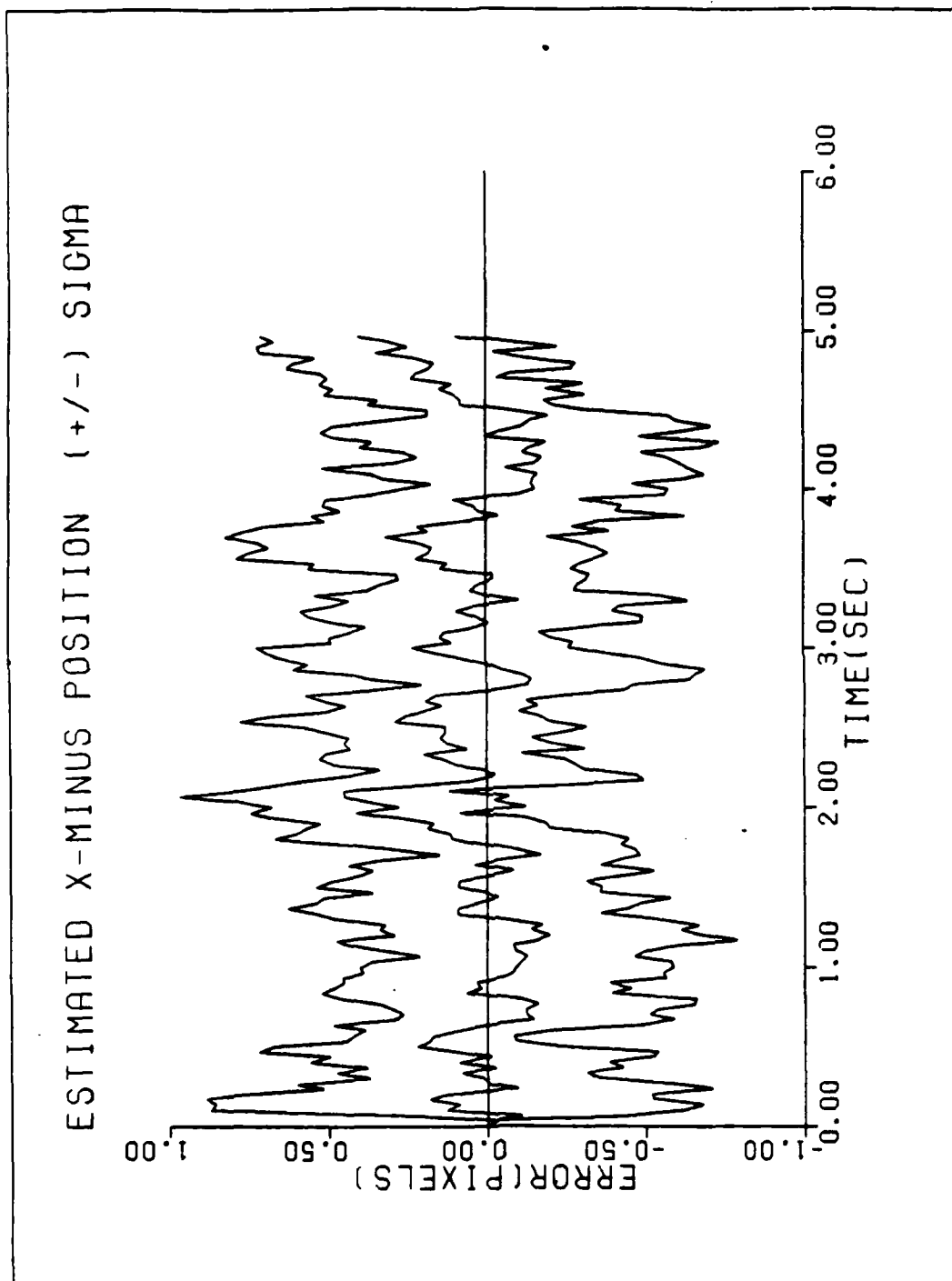


Figure E9 Performance plot for T1AR.2SA

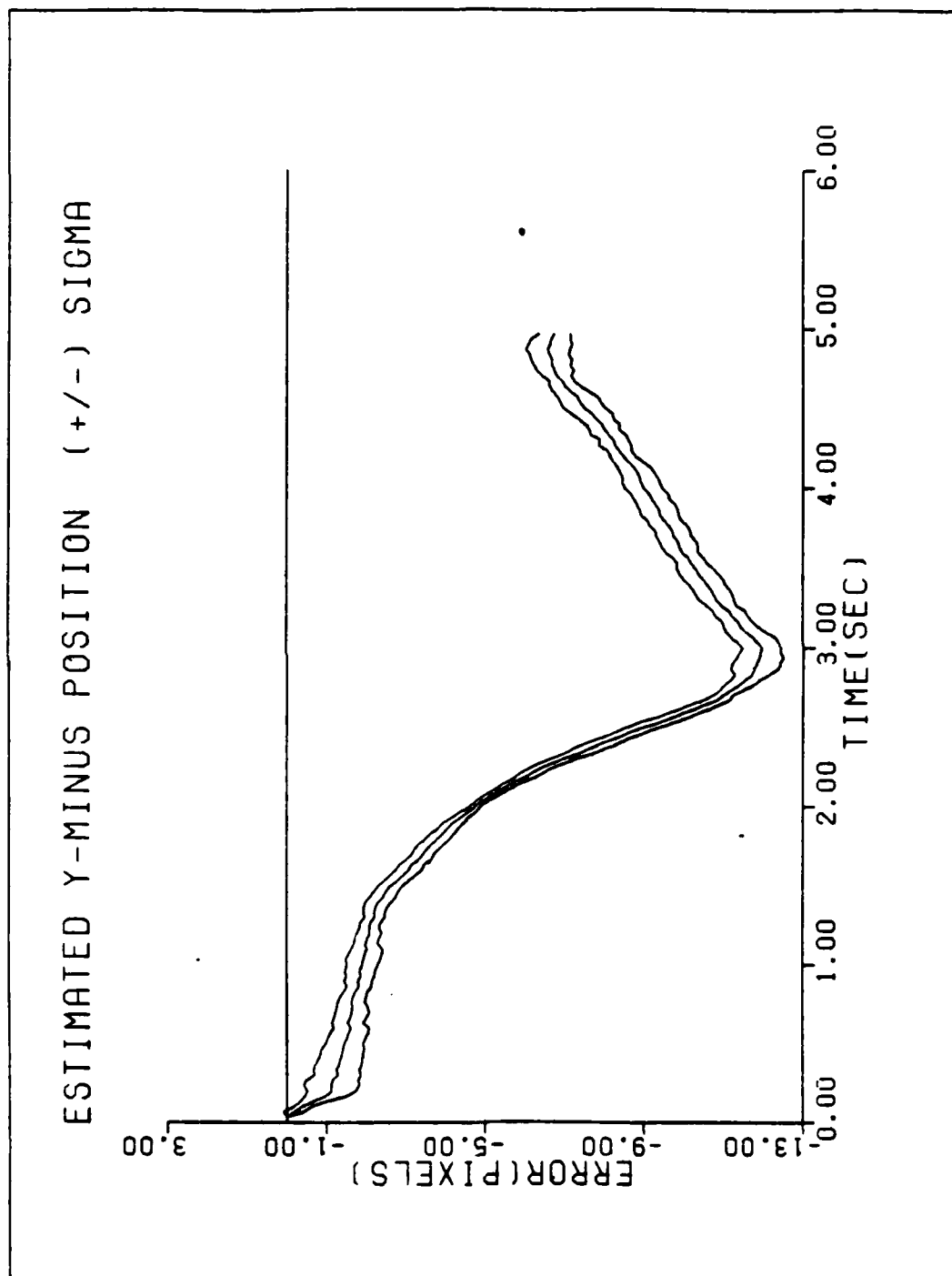


Figure E10 Performance plot for TIAR.2SA

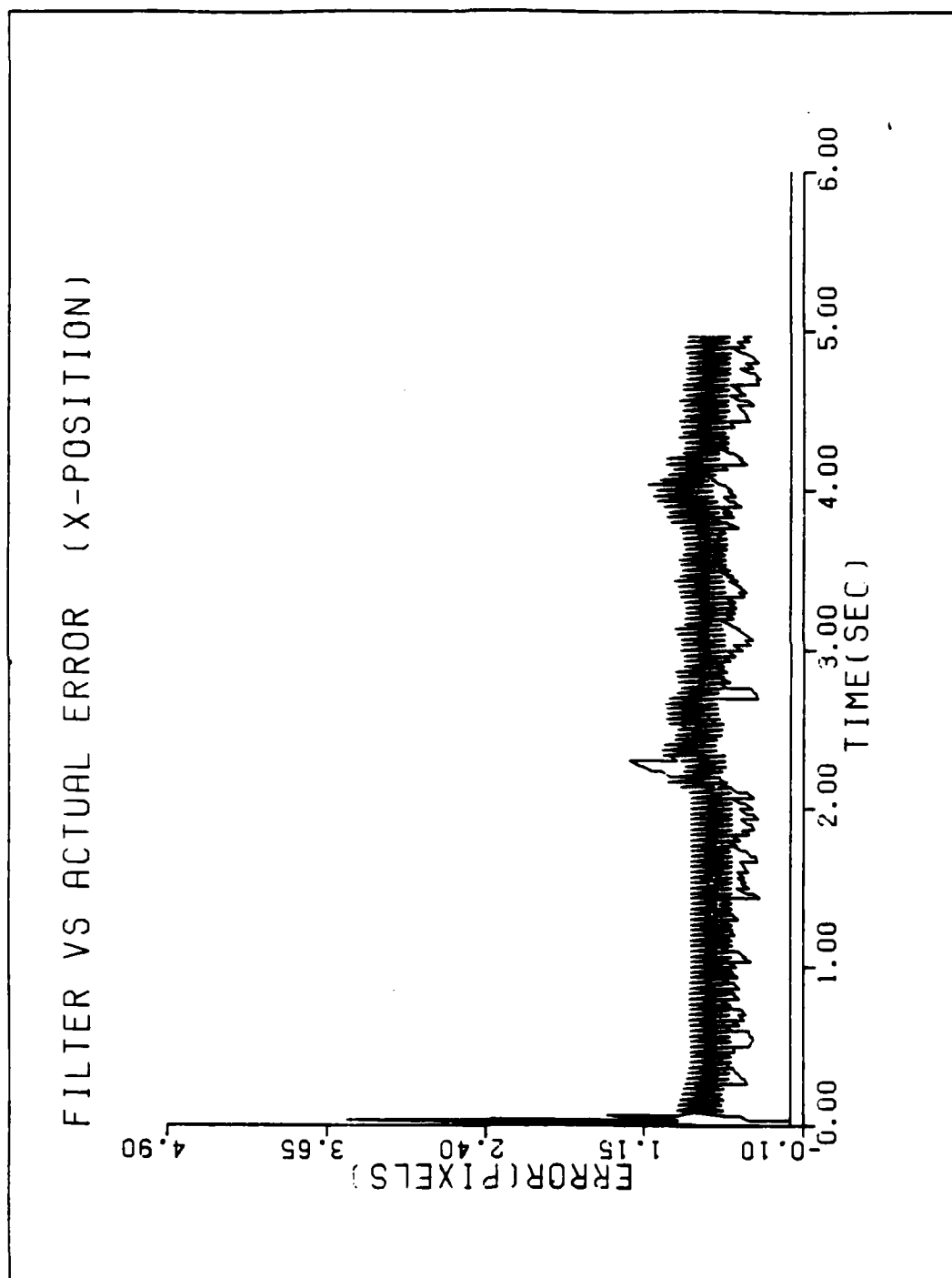


Figure E11 Performance plot for T2G10AR5

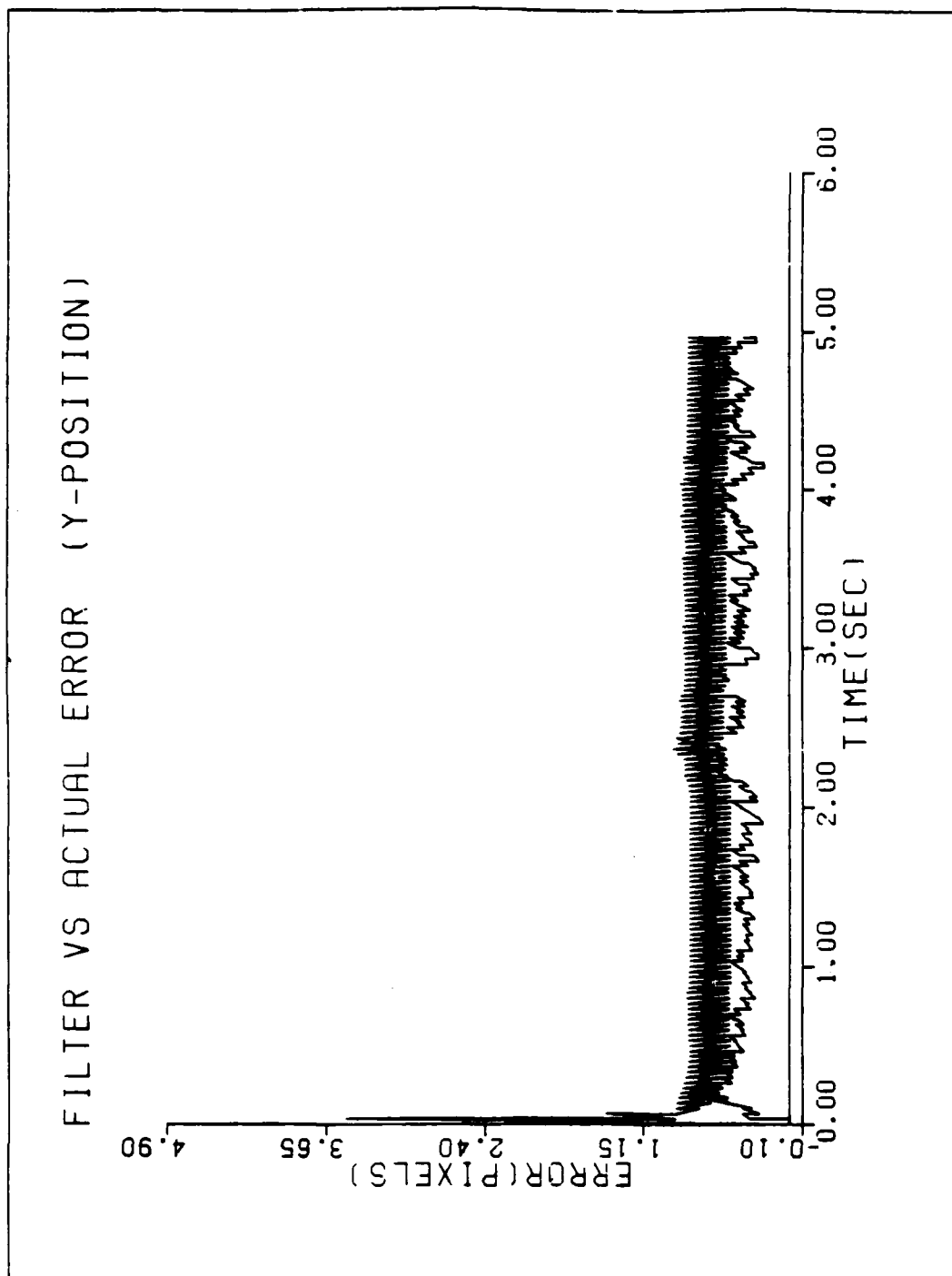


Figure E12 Performance plot for T2G10AR5

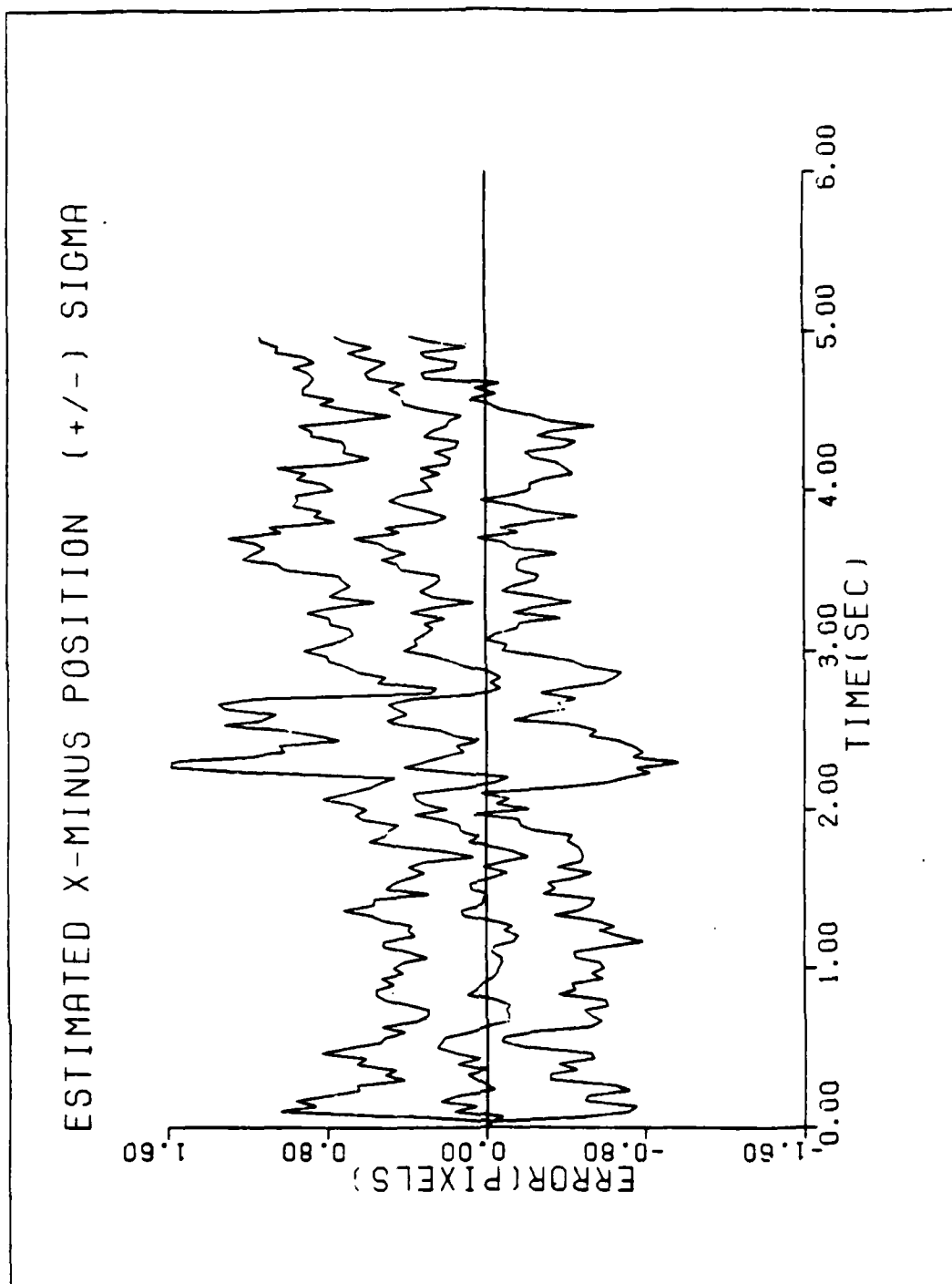


Figure E13 Performance plot for T2G10AR5

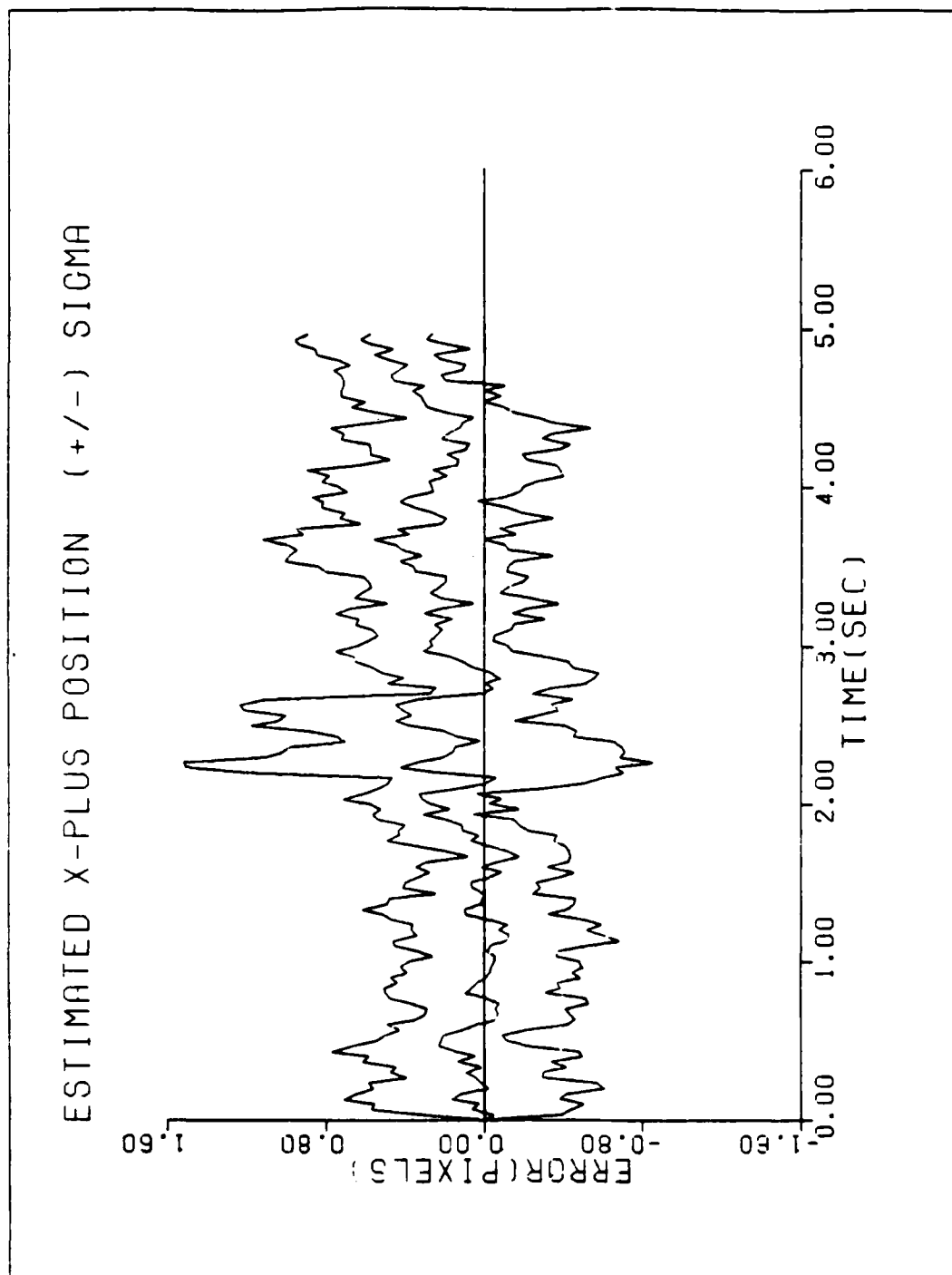


Figure E14 Performance plot for T2G10AR5

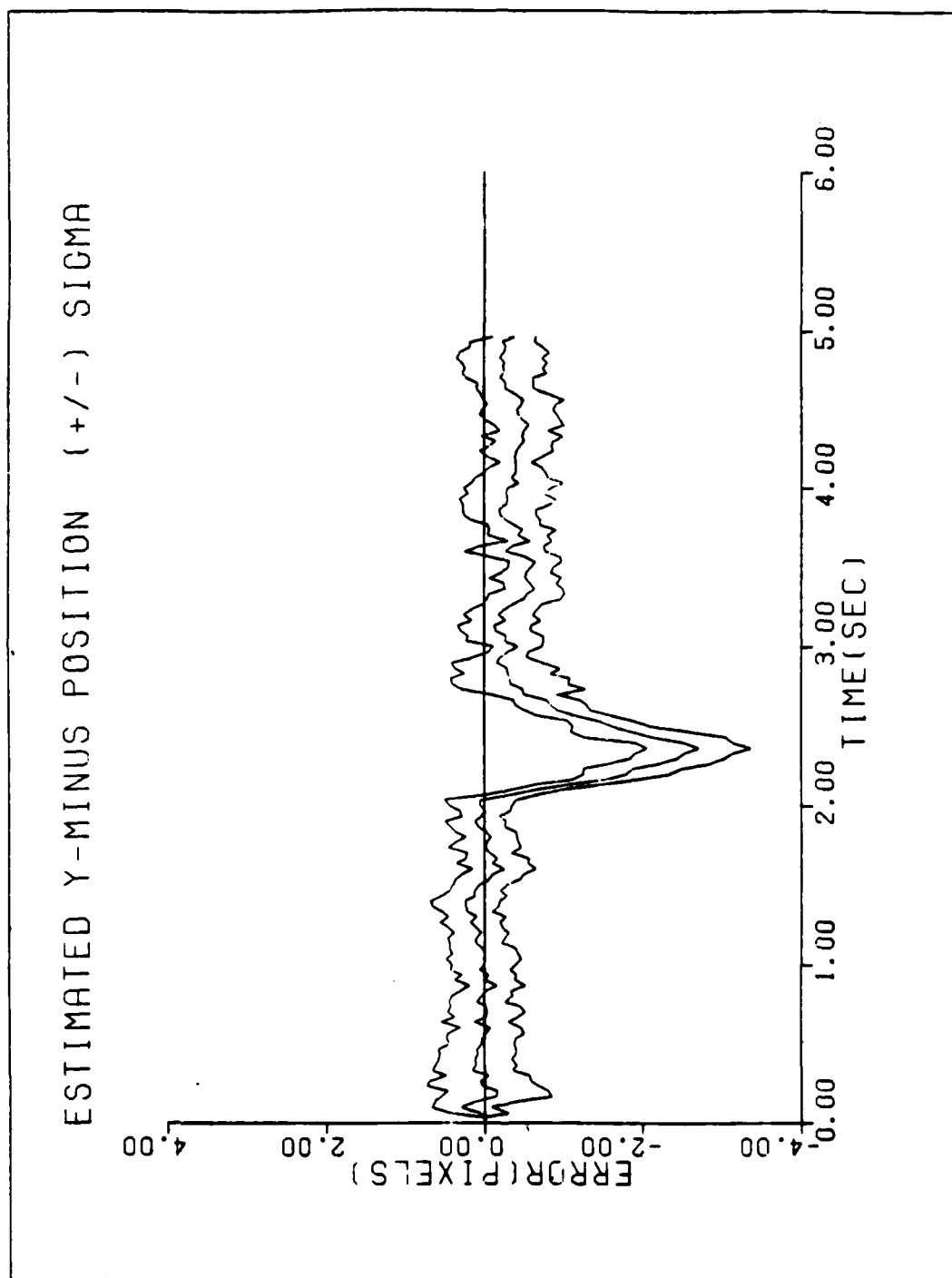


Figure E15 Performance plot for T2G10AR5

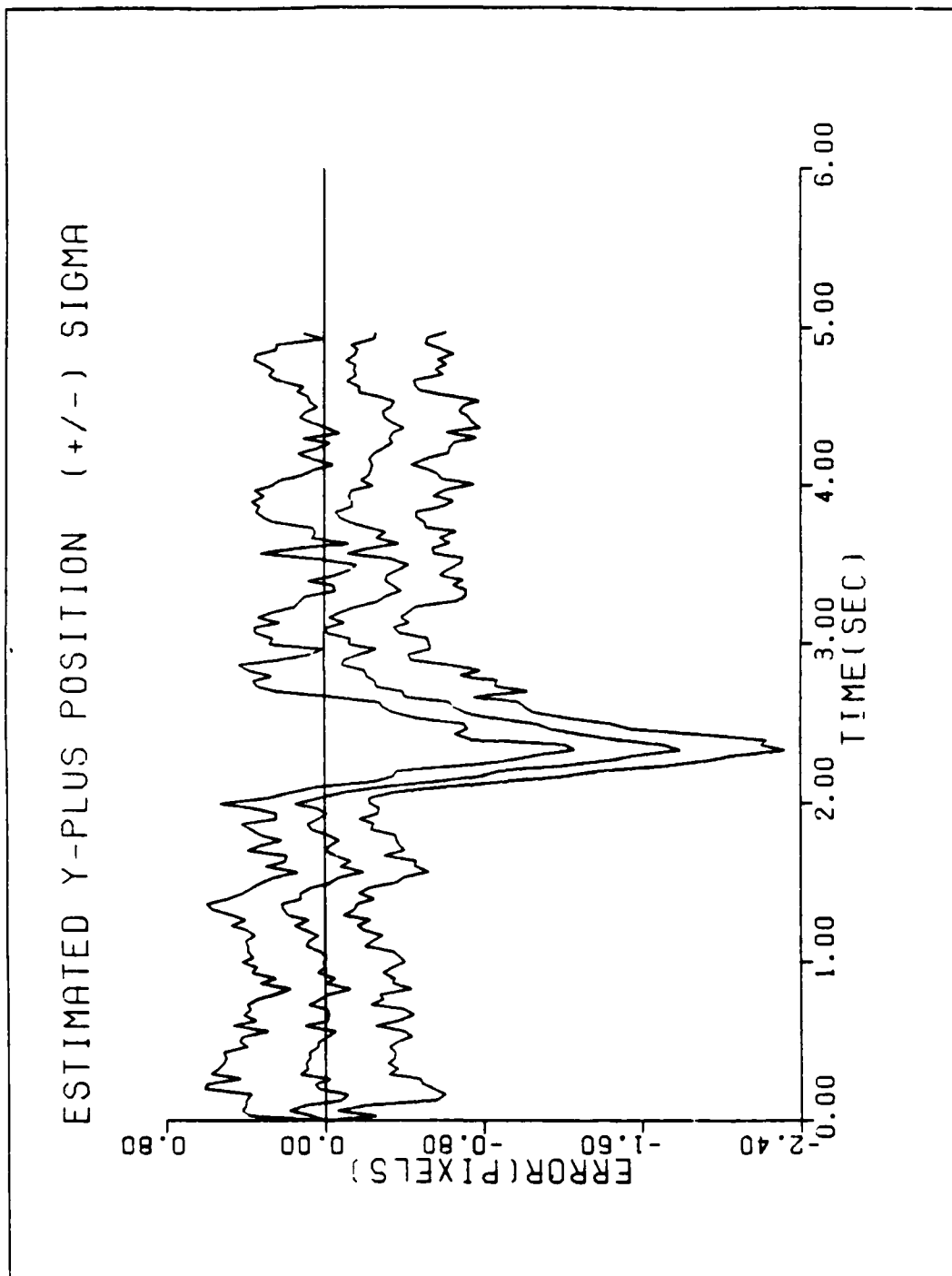


Figure E16 Performance plot for T2G10AR5

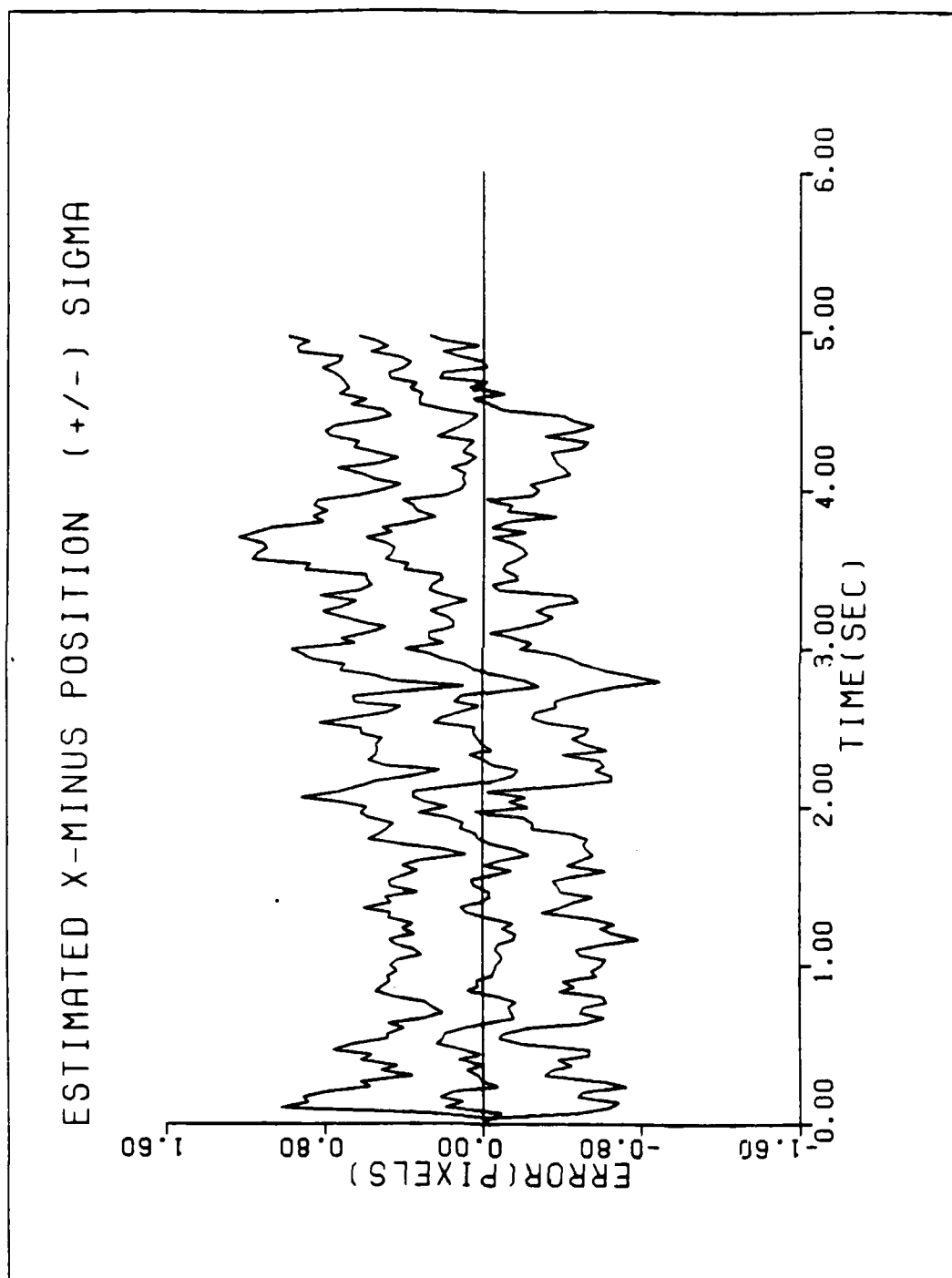


Figure E17 Performance plot for T2G10BL*

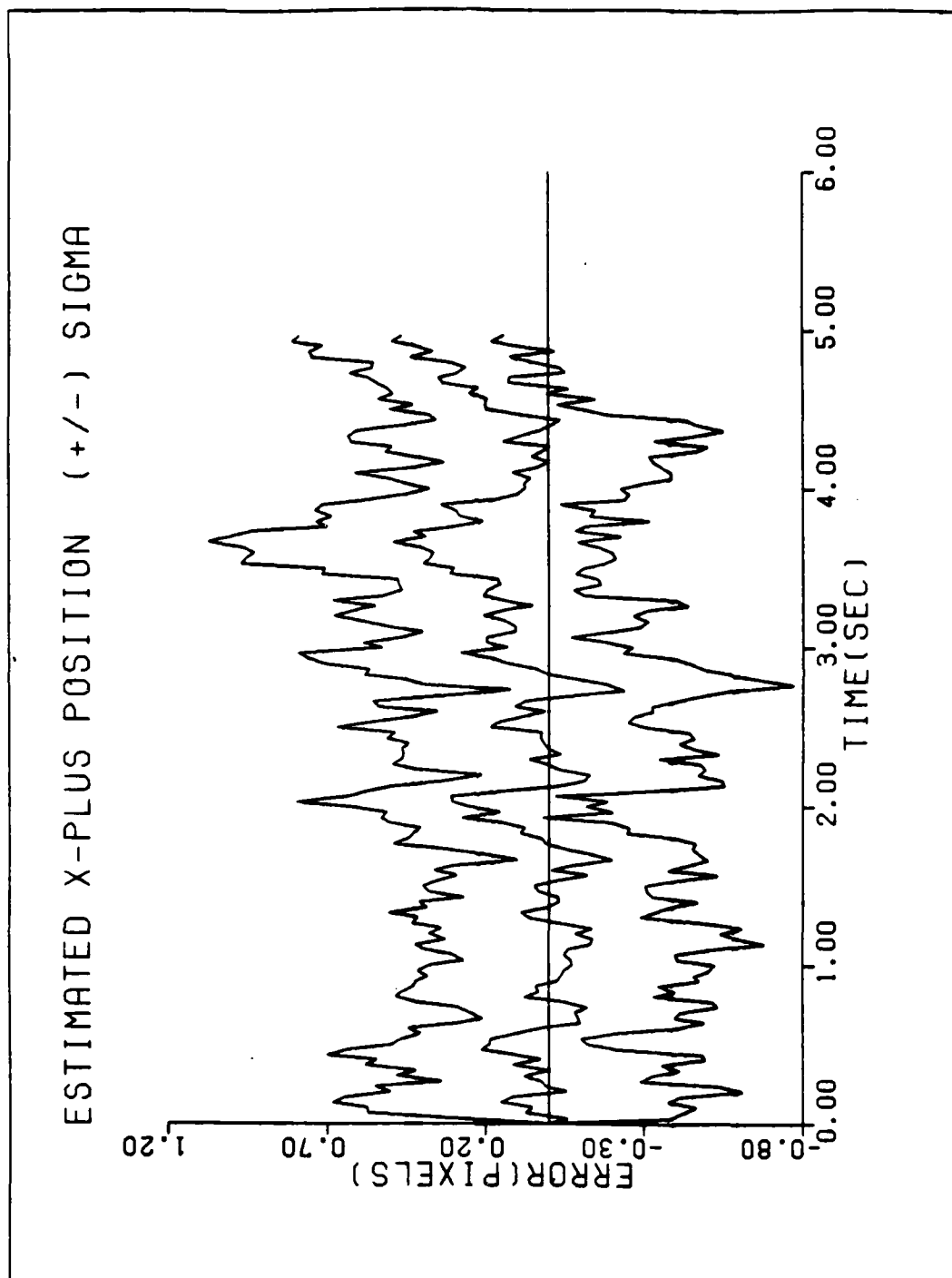


Figure E18 Performance plot for T2G10BL*

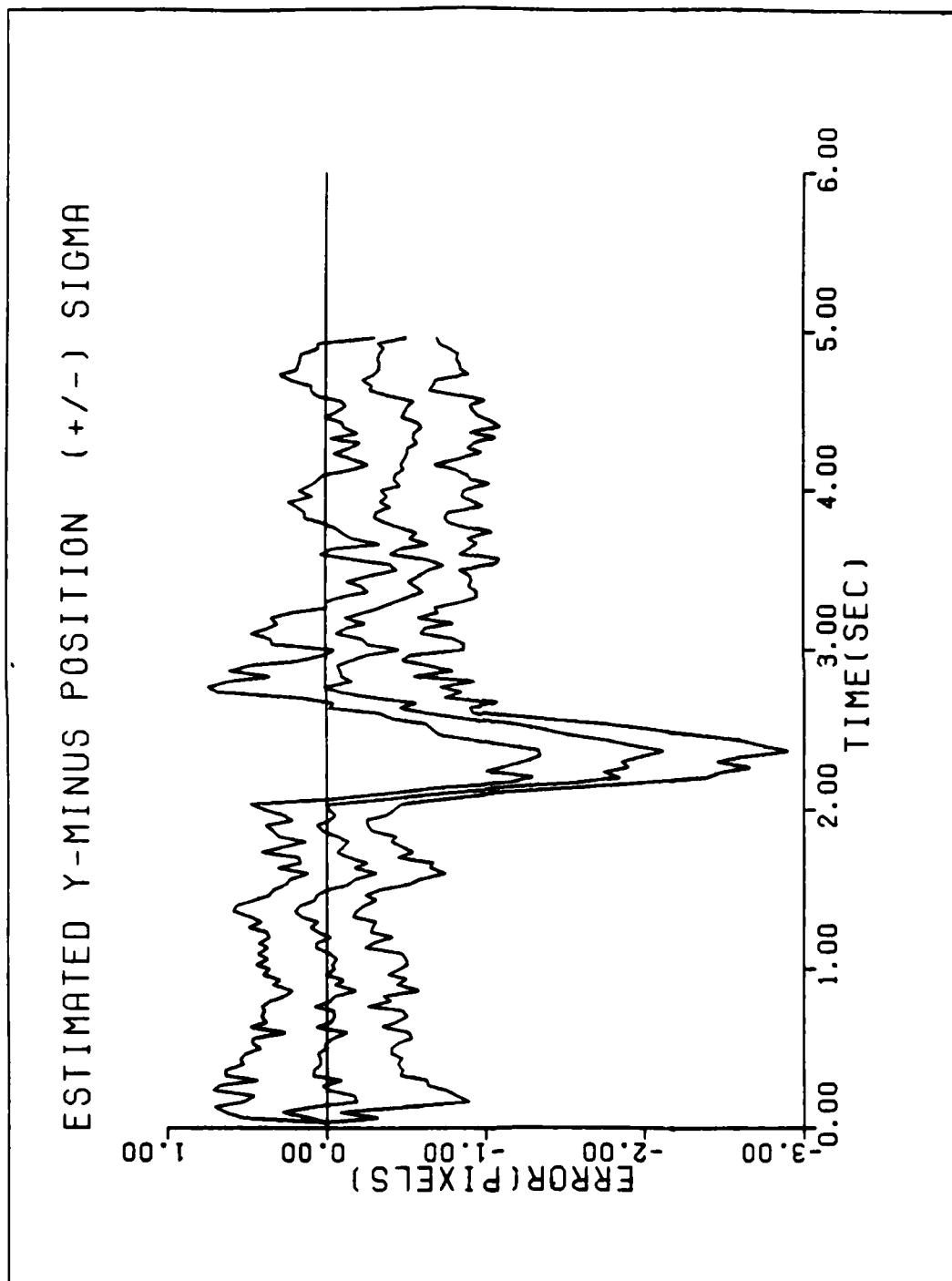


Figure E19 Performance plot for T2G10BL*

ESTIMATED Y-PLUS POSITION (+/-) SIGMA

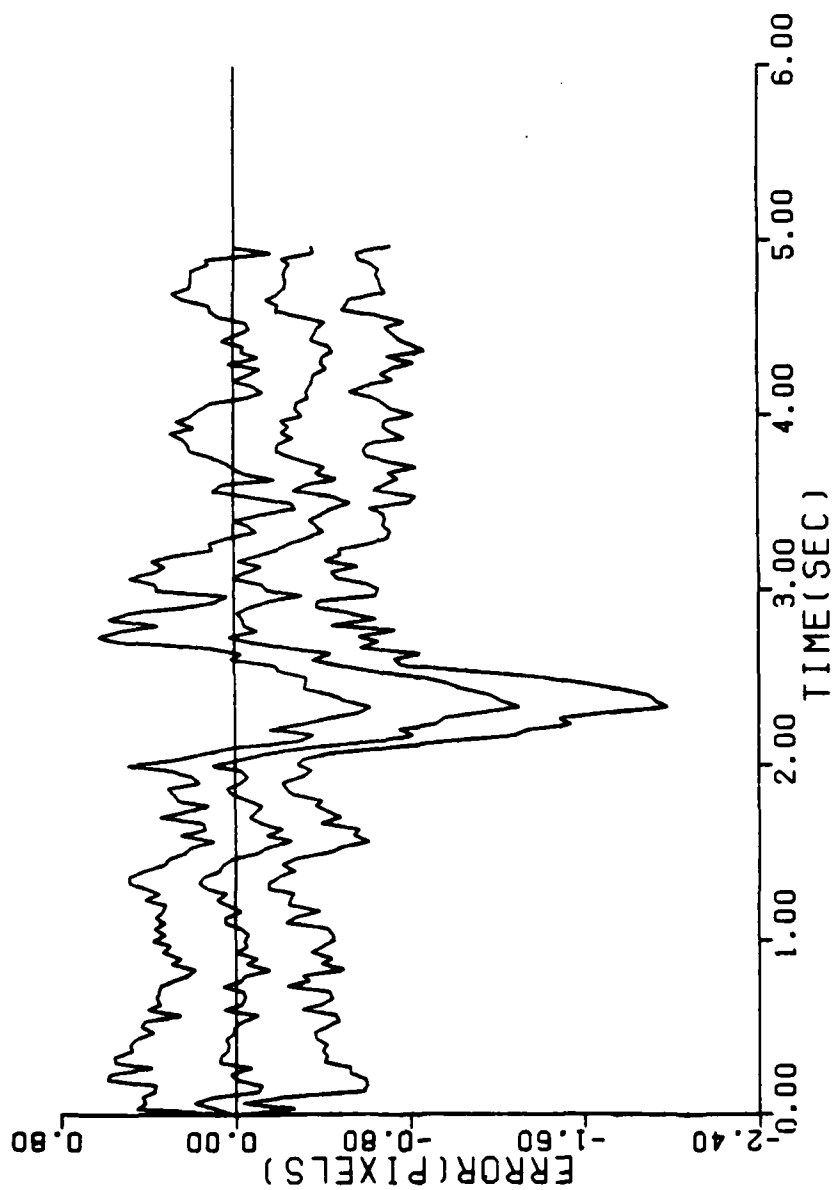


Figure E20 Performance plot for T2G10BL*

ESTIMATED X-MINUS POSITION (+/-) SIGMA

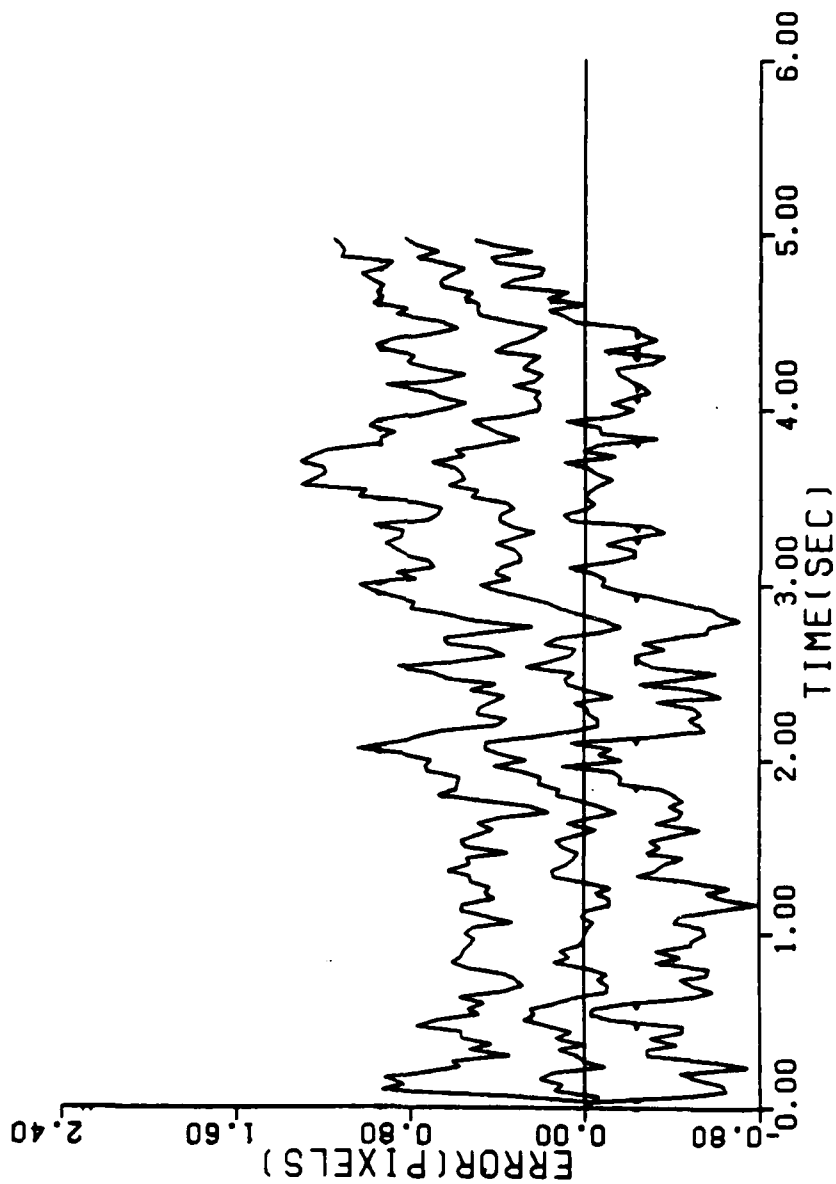


Figure E21 Performance plot for T2G10D10*

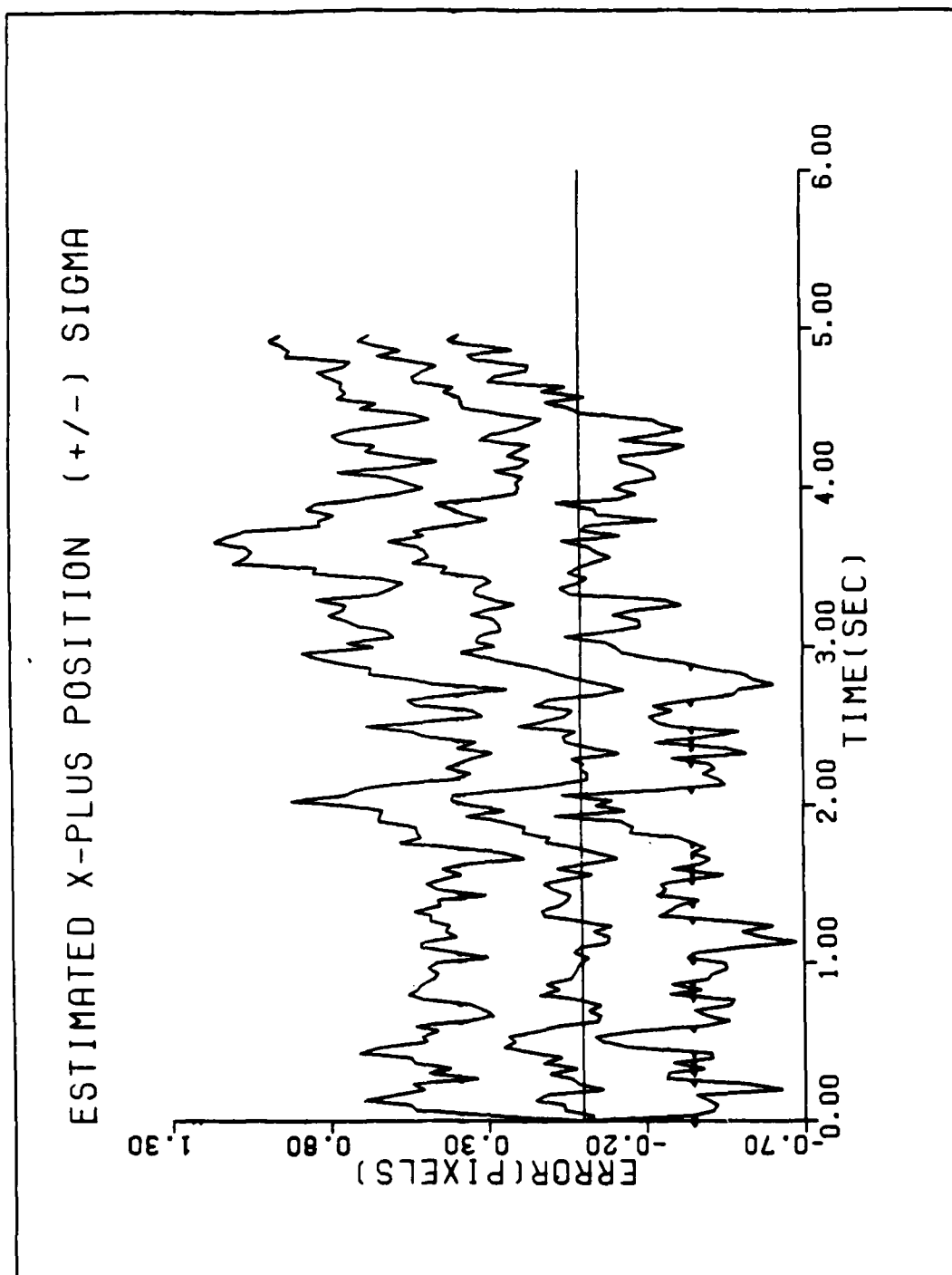


Figure E22 Performance plot for T2G10D10*

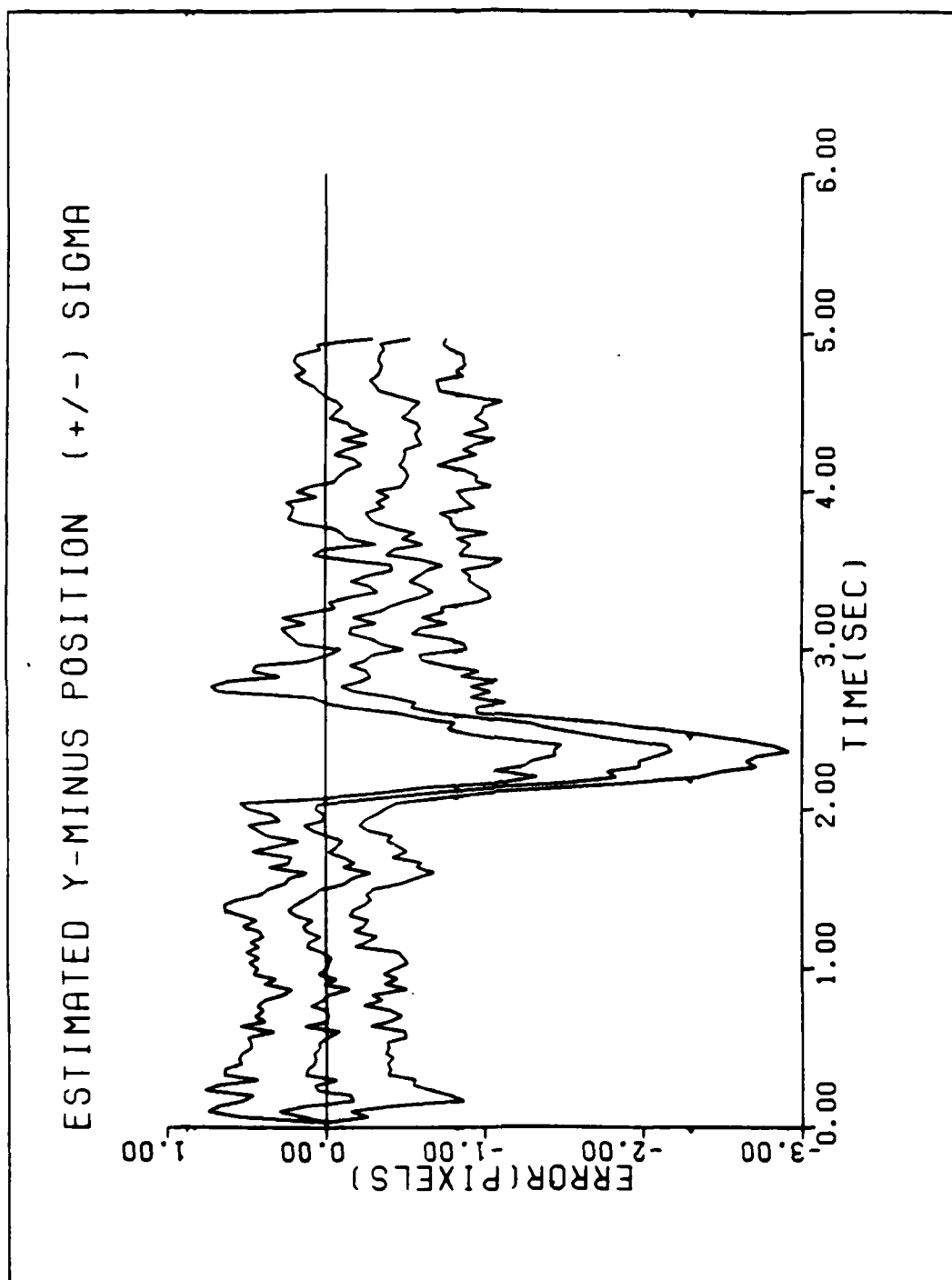


Figure E23 Performance plot for T2G10D10*

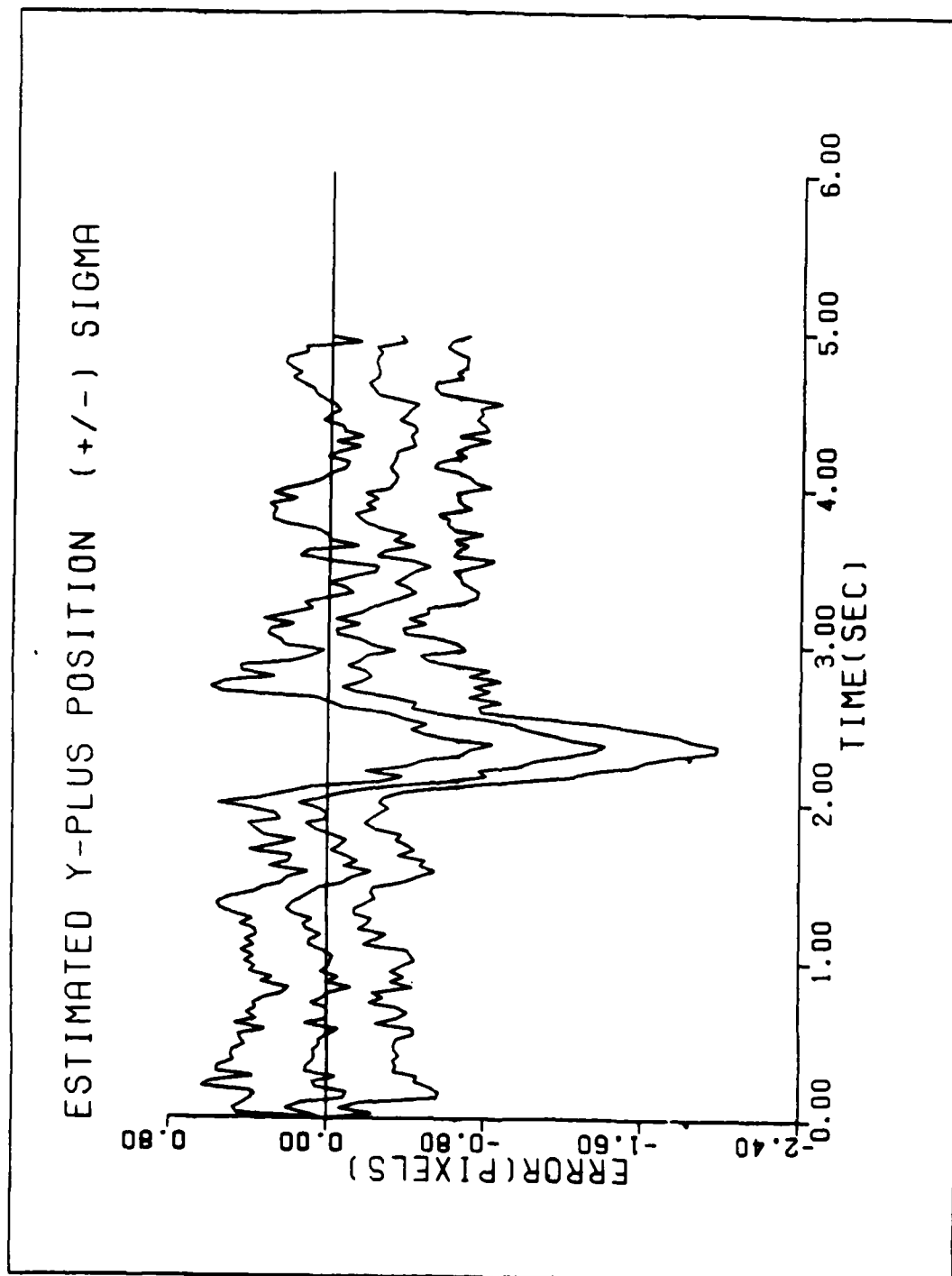


Figure E24 Performance plot for T2G10MD*

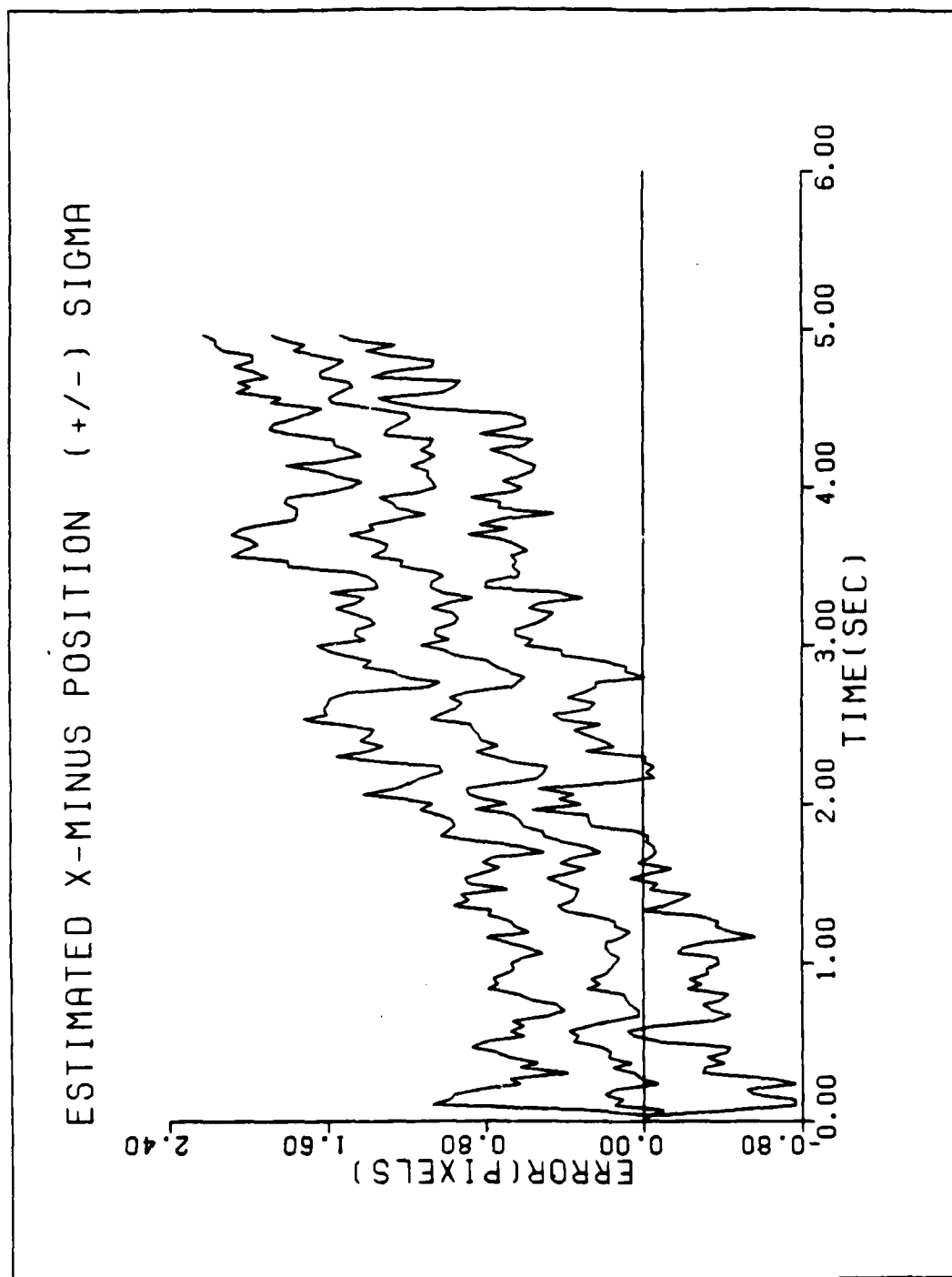


Figure E25 Performance plot for T2G10MD*SPN

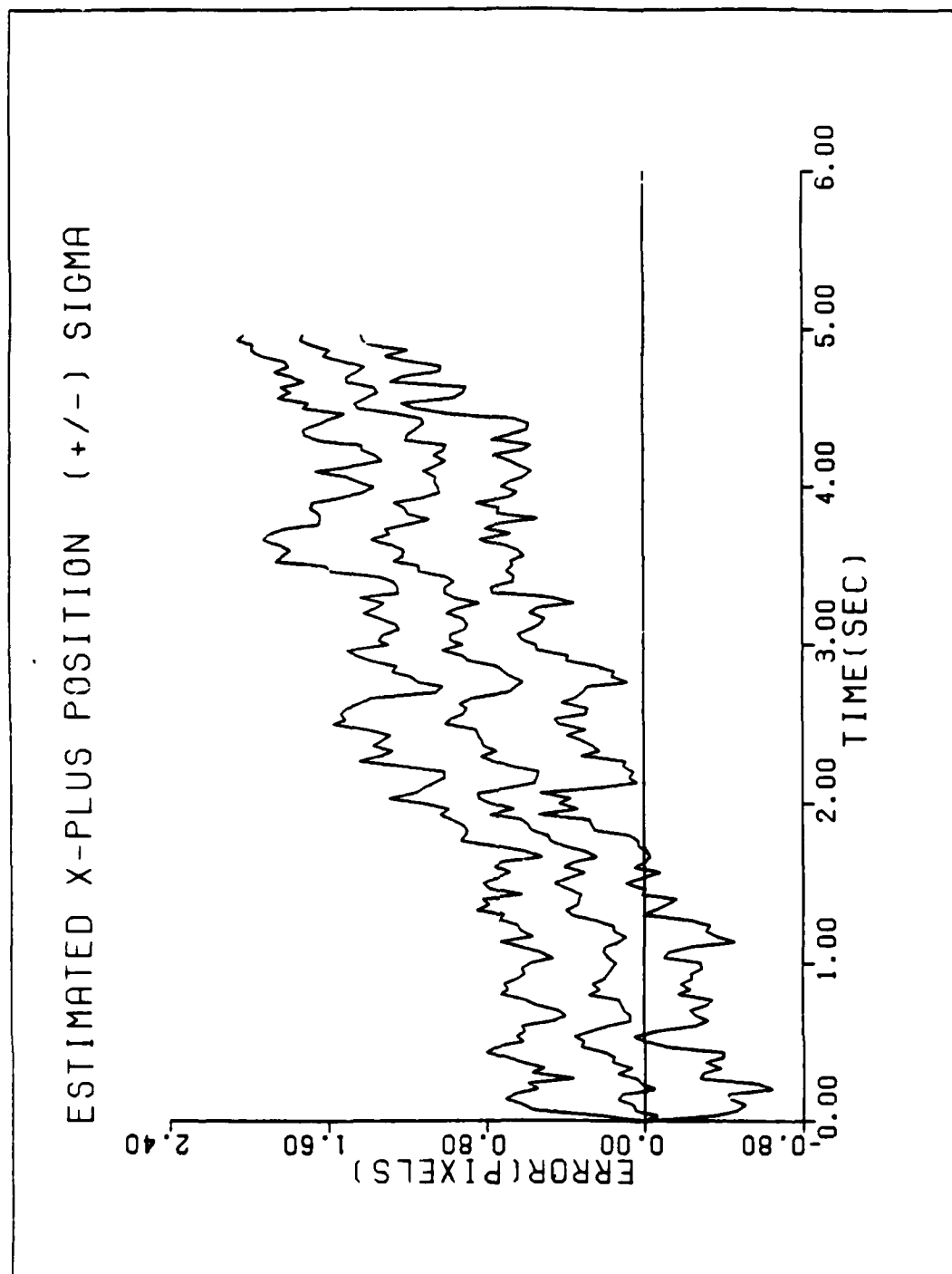


Figure E26 Performance plot for T2G10MD*SPN

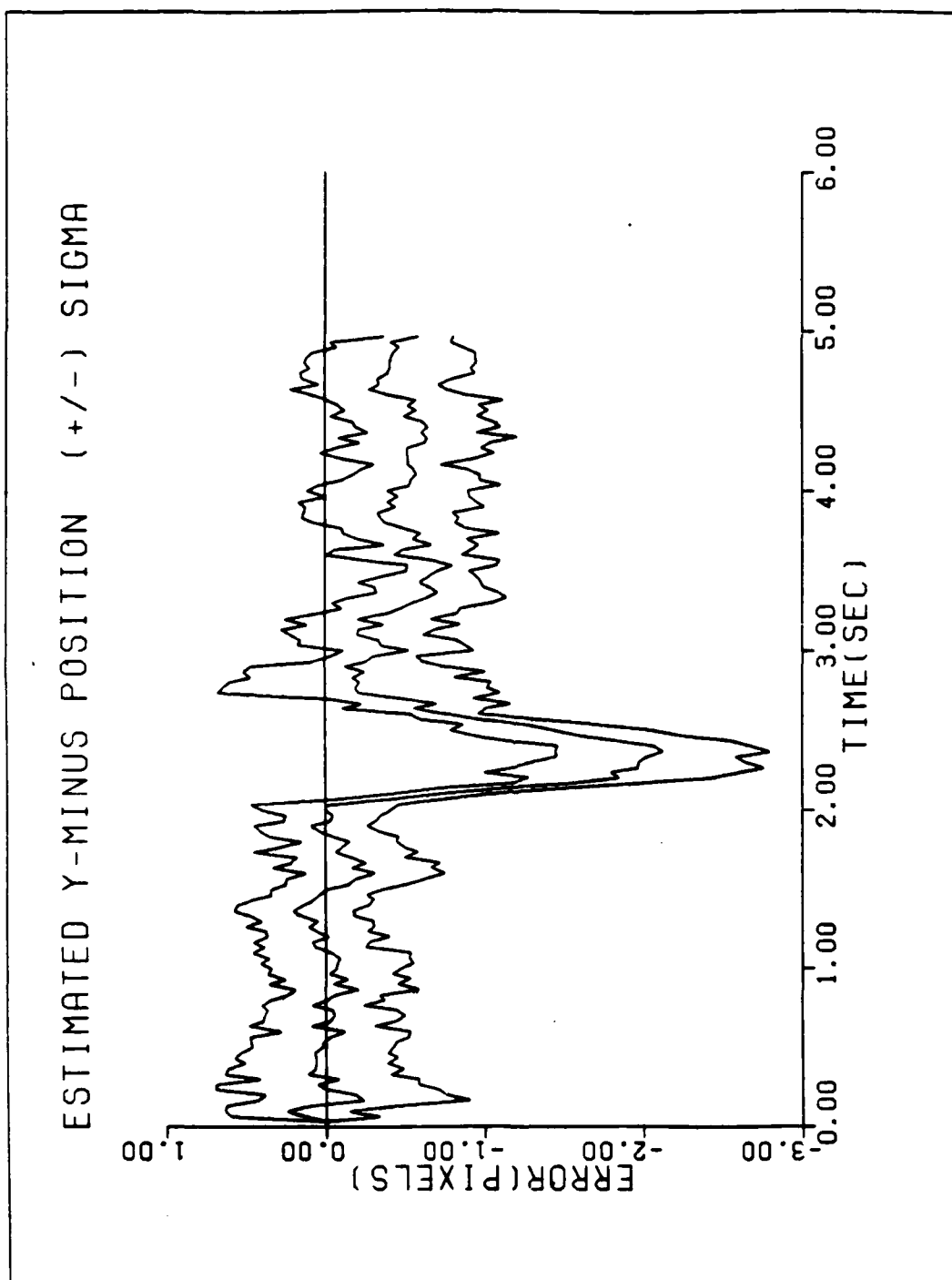


Figure E27 Performance plot for T2G10MD*SPN

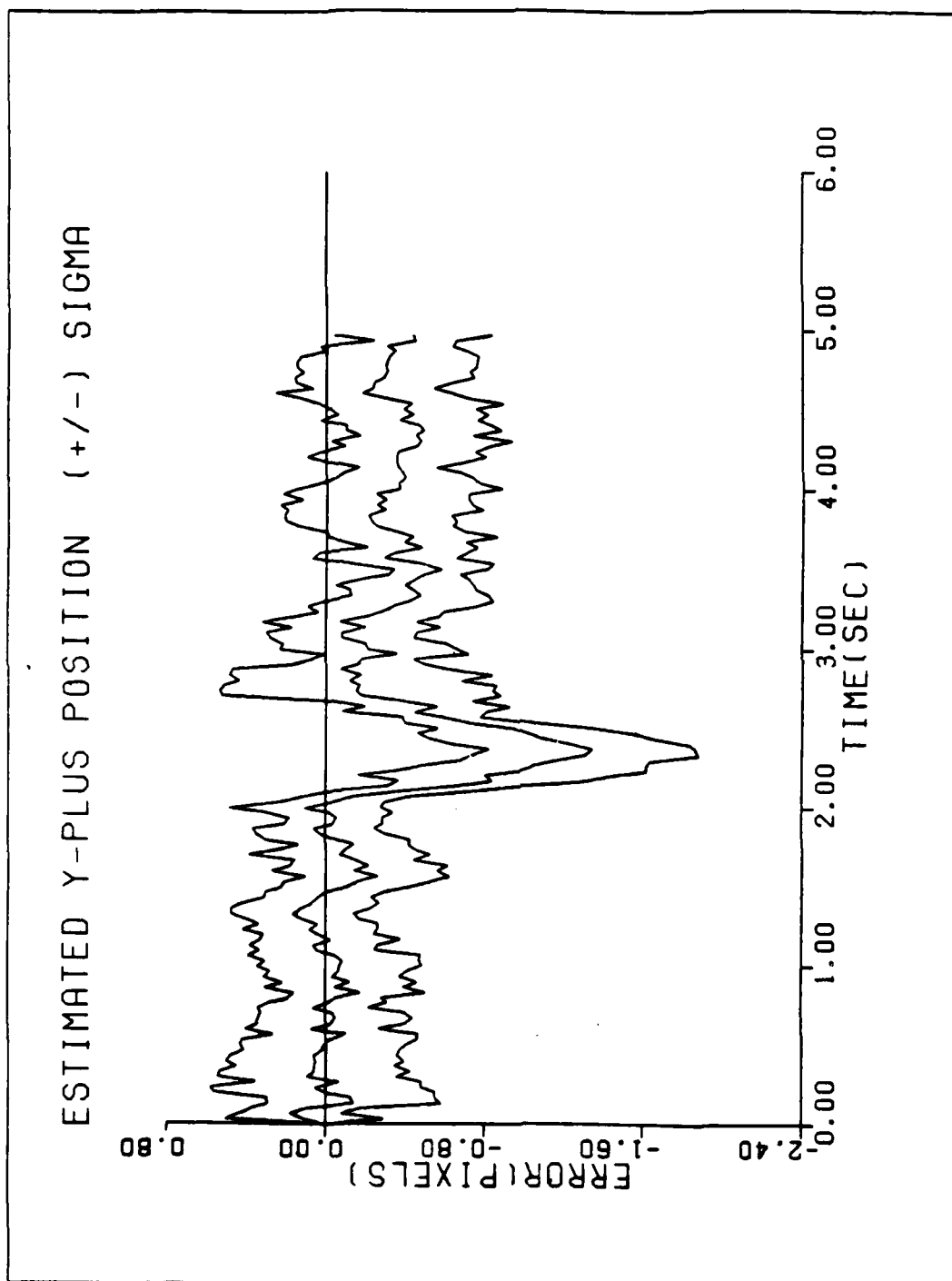


Figure E28 Performance plot for T2G10MD*SPN

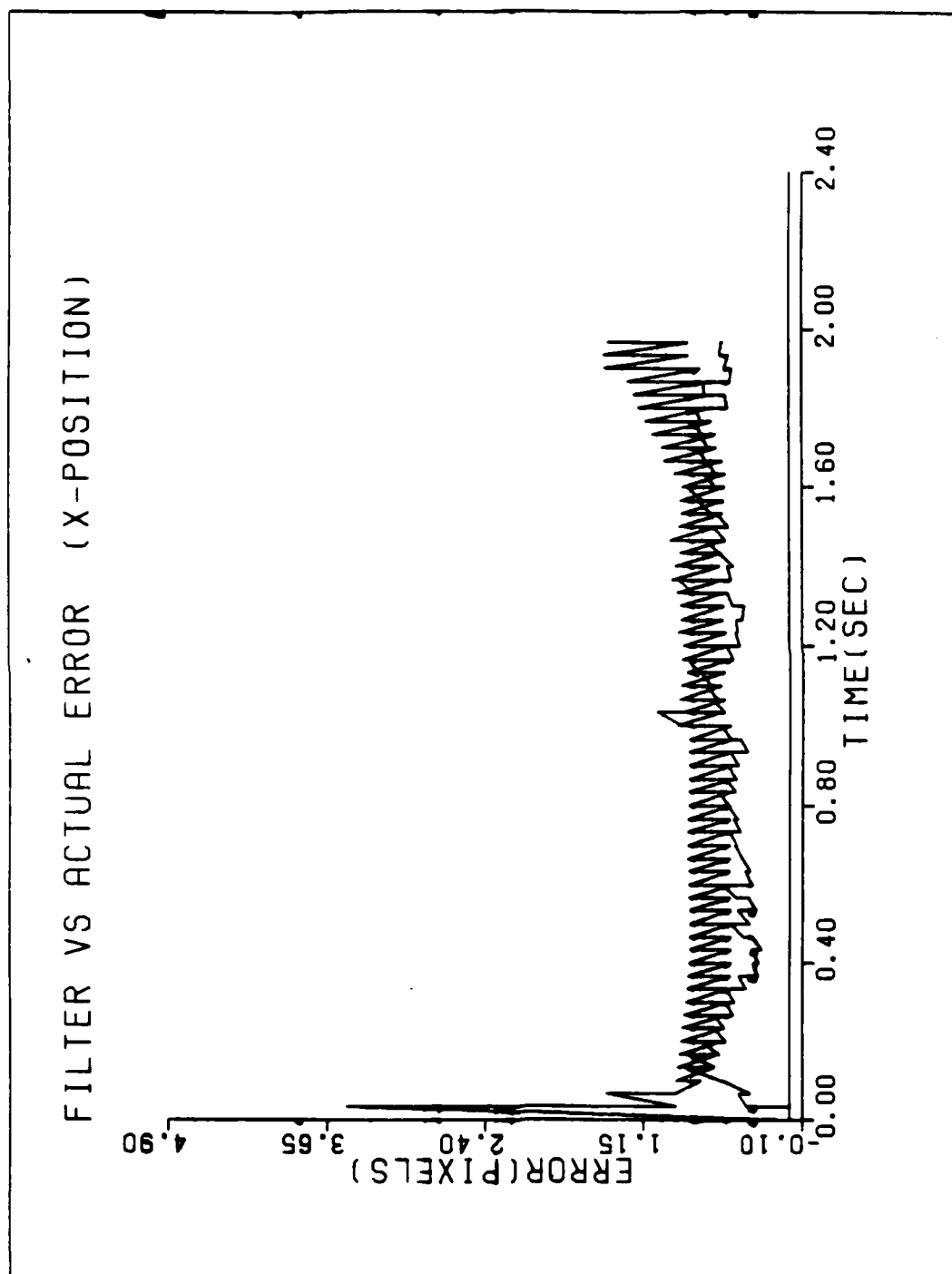


Figure E29 Performance plot for TARGET/DECOY

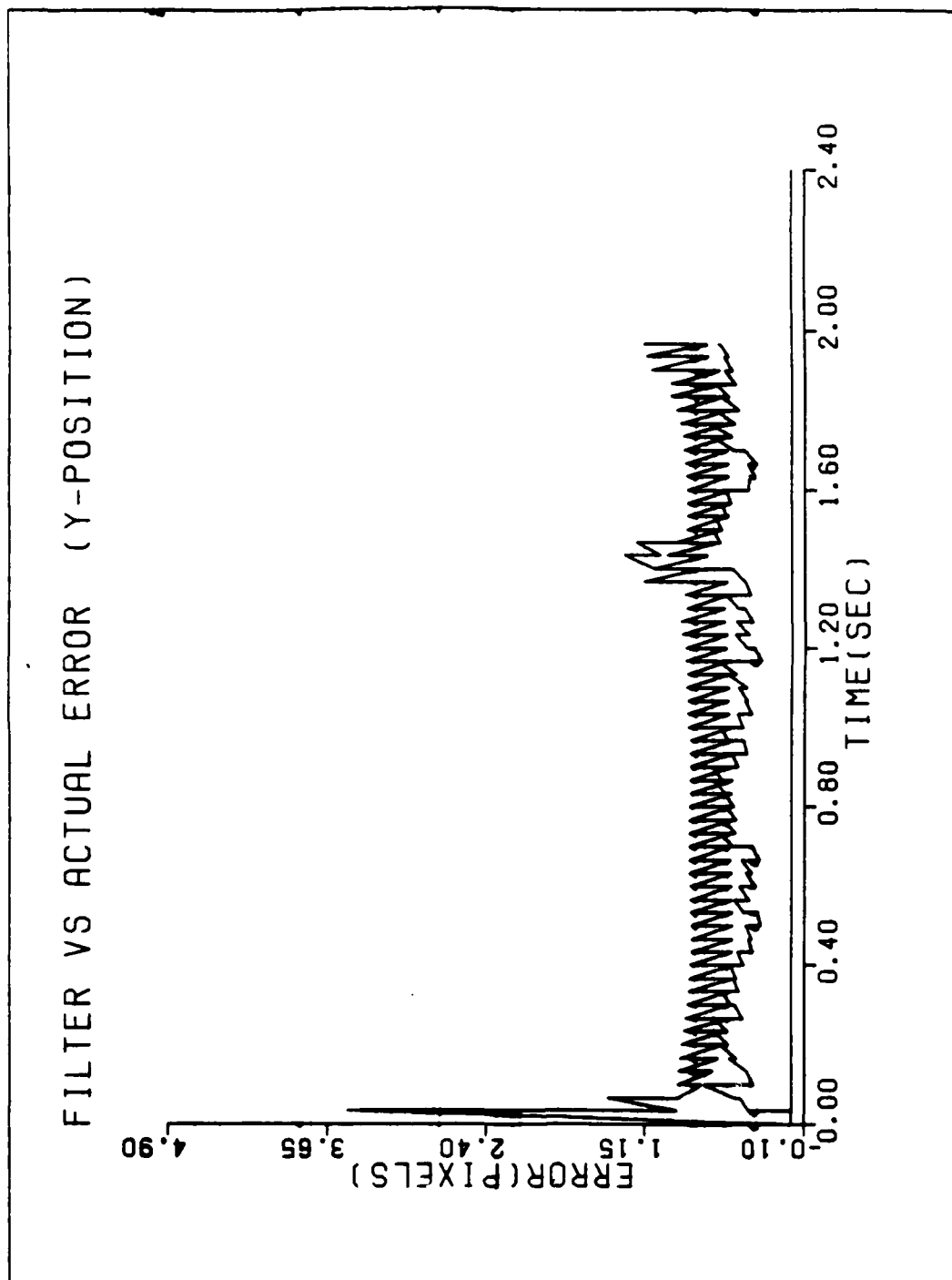


Figure E30 Performance plot for TARGET/DECOY

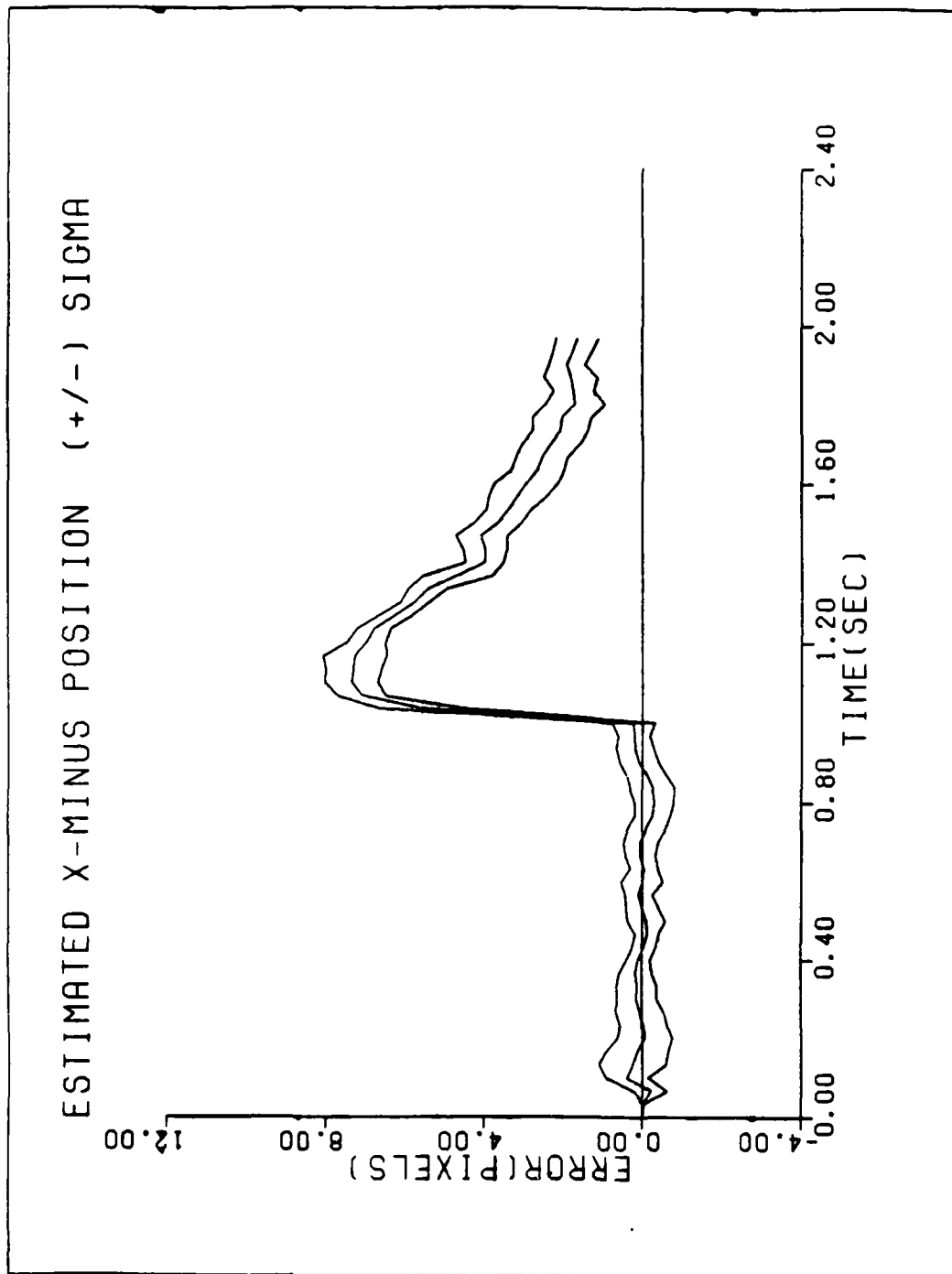


Figure E31 Performance plot for TARGET/DECOY

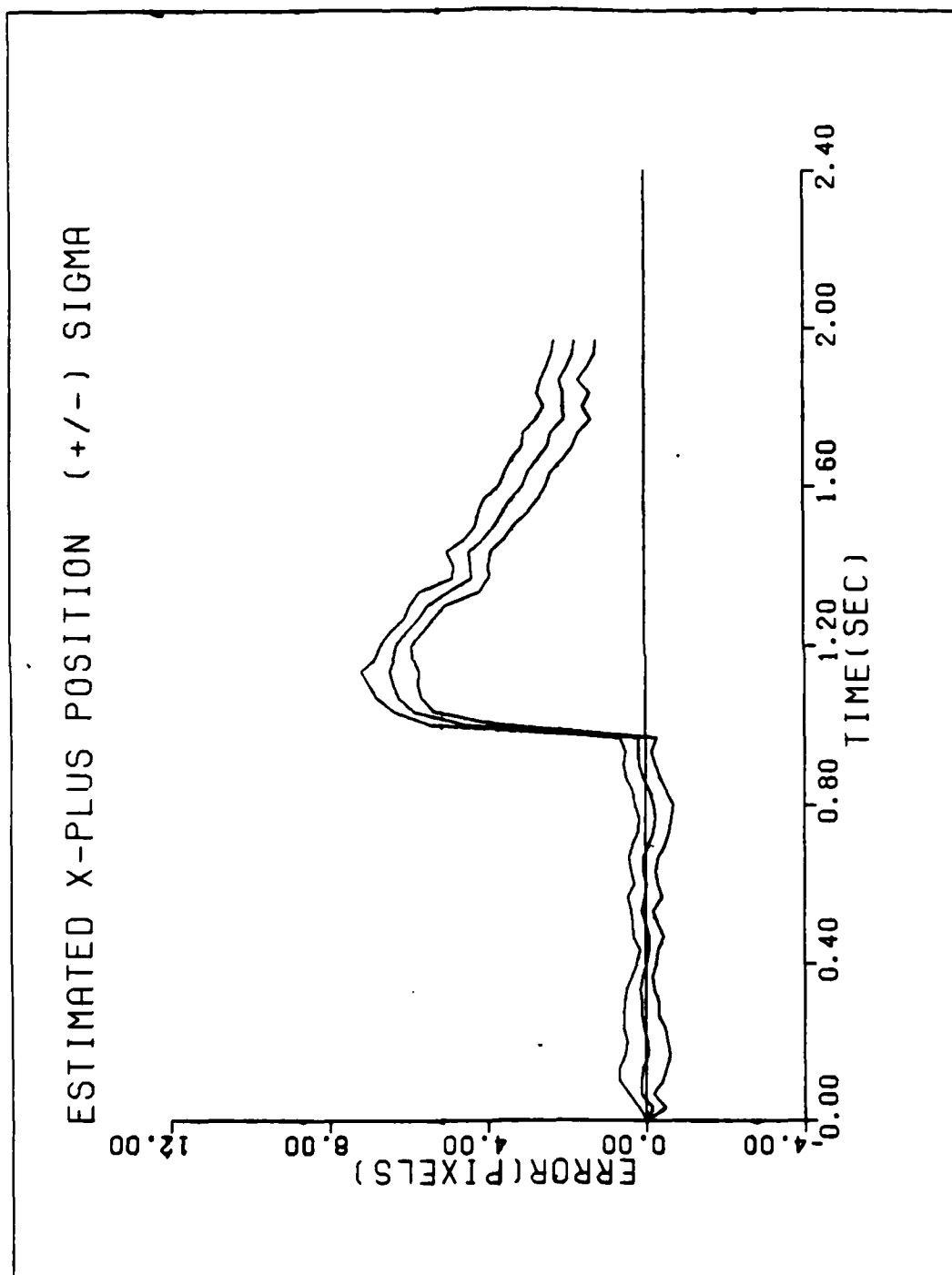


Figure E32 Performance plot for TARGET/DECOY

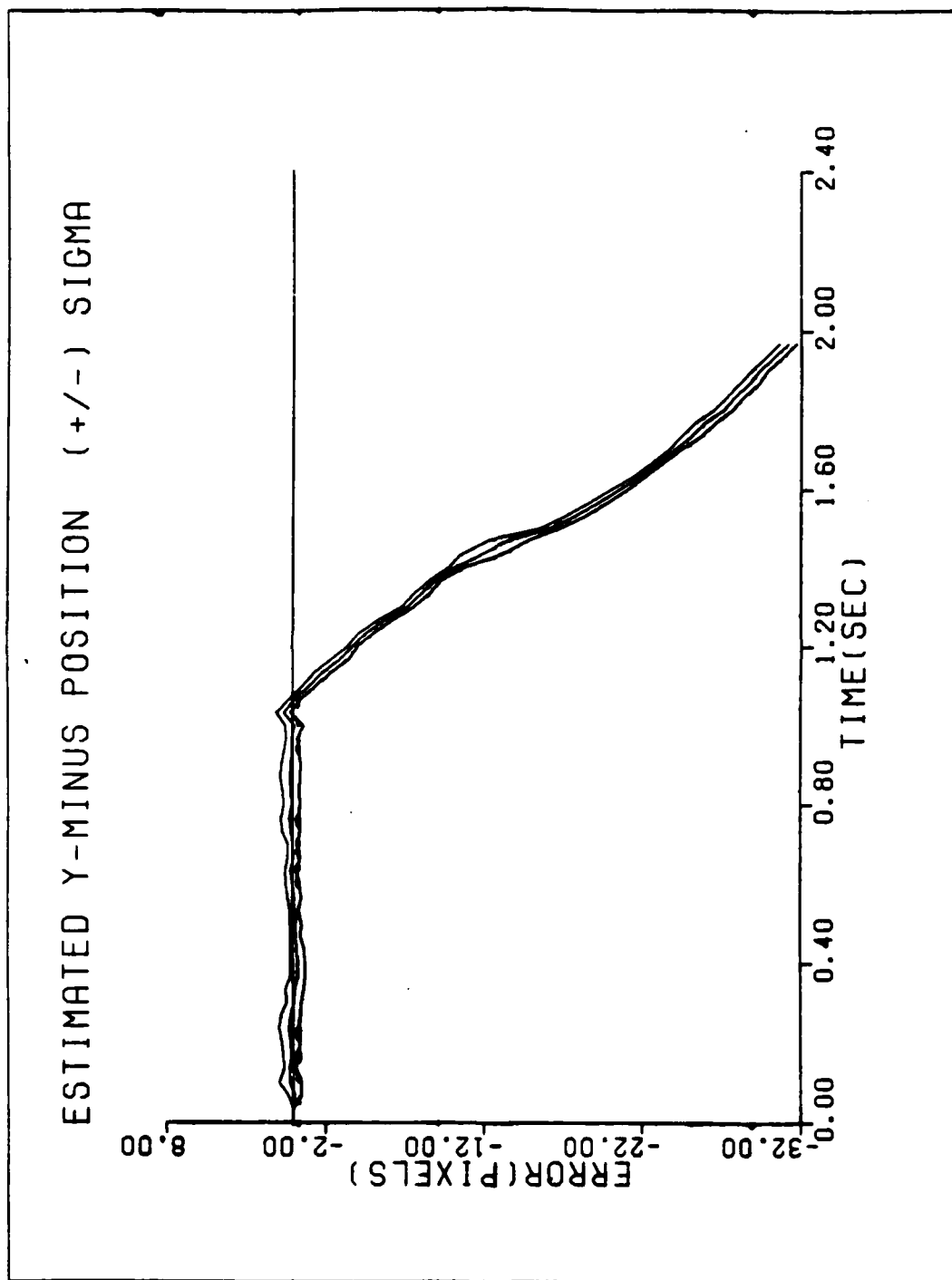


Figure E33 Performance plot for TARGET/DECOY

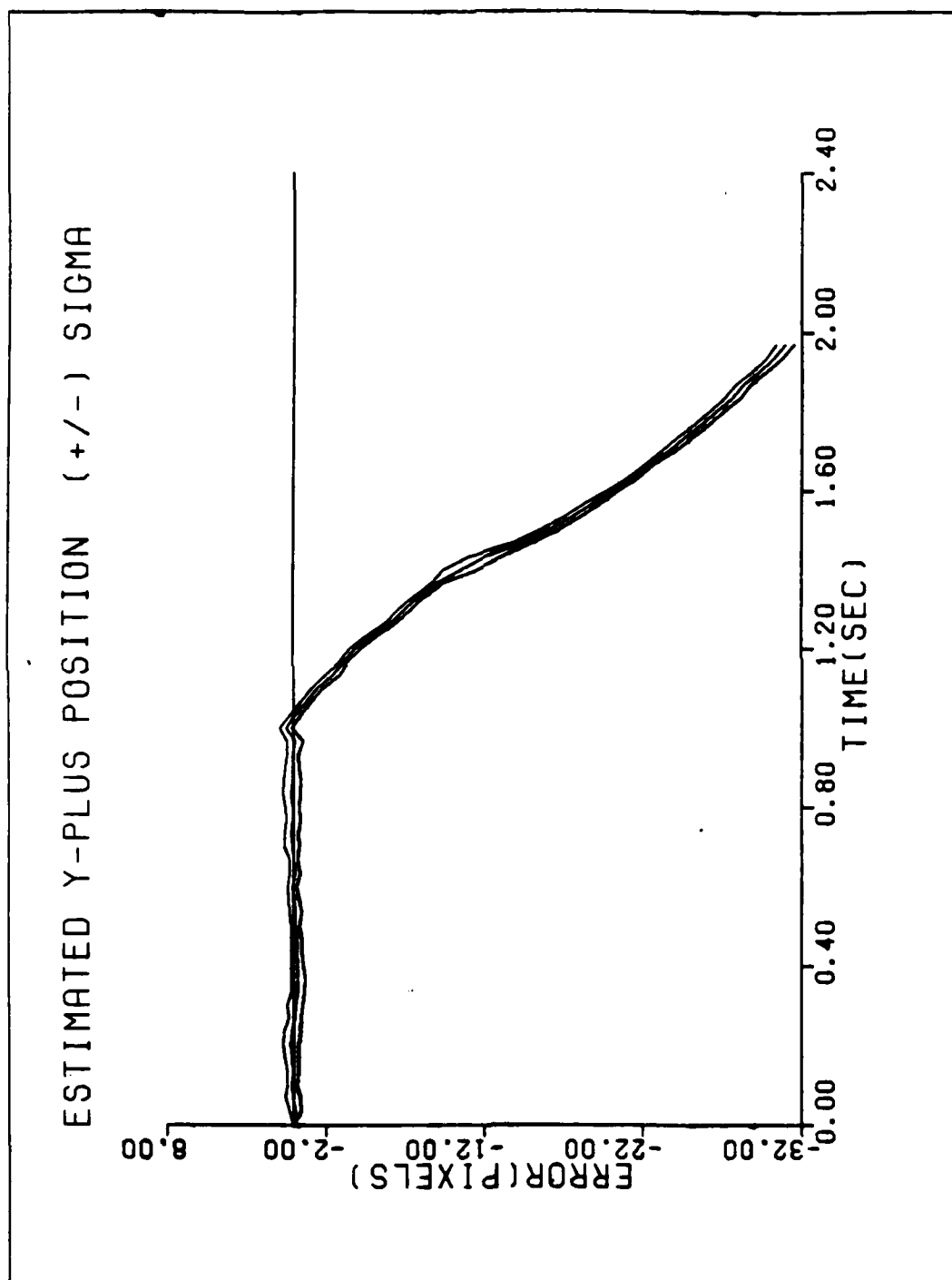


Figure E34 Performance plot for TARGET/DECOY

Bibliography

1. Flynn, Patrick M., "Alternative Dynamics Models and Multiple Model Filtering for a Short Range Tracker," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1981.
2. Harnly, Douglas A. and Robert L. Jensen, "An Adaptive Distributed-Measurement Extended Kalman Filter for a Short Range Tracker," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1979.
3. Kozemchak, Mark R., "Enhanced Image Tracking: Analysis of Two Acceleration Models in Tracking Multiple Hot-Spot Images," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1982.
4. Loving, Phyllis A., "Bayesian vs MAP Multiple Model Adaptive Estimation for Field of View Expansion in Tracking Airborne Targets," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, March 1985.
5. Maybeck, Peter S. Stochastic Models, Estimation, and Control Volume I, New York: Academic Press Incorporated, 1979.
6. Maybeck, Peter S. Stochastic Models, Estimation, and Control Volume II, New York: Academic Press Incorporated, 1982.
7. Maybeck, P.S., D.A. Harnly and R.L. Jensen, "Robustness of a New Infrared Target Tracker," Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, Ohio, pp 639-644, May 1980.
8. Maybeck, P.S., R.L. Jensen, and D.A. Harnly, "An Adaptive Extended Kalman Filter for Target Image Tracking," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-17, pp 173-180, March 1981.
9. Maybeck, P.S., and D.E. Mercier, "A Target Tracker Using Spatially Distributed Infrared Measurements," IEEE Transactions on Automatic Control, Vol. AC-25, No. 2, pp 222-225, April 1980.
10. Maybeck, P.S., and S.K. Rogers, "Adaptive Tracking of Multiple Hot-Spot Target IR Images," IEEE Transactions on Automatic Control, Vol. AC-28, No. 10, pp 937-943, October 1983.

11. Maybeck, P.S., W.H. Worsley and P.M. Flynn, "Investigation of Constant Turn-Rate Dynamics Models in Filters for Airborne Vehicle Tracking," Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, Ohio, pp 896-903, May 1982.
12. Mercier, D.E., "An Extended Kalman Filter for Use in a Shared Aperture Medium Range Tracker," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1978.
13. Millner, Paul P., "Enhanced Tracking of Airborne Targets Using a Correlator/Kalman Filter," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1982.
14. Rogers, Steven K., "Enhanced Tracking of Airborne Targets Using Forward Looking Infrared Measurements," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1981.
15. Singletery, James Jr., "Adaptive Laser Pointing and Tracking Problem," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1980.
16. Suizu, Robert I., "Enhanced Tracking of Airborne Targets Using Multiple Model Filtering Techniques for Adaptive Field of View Expansion," M.S. Thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1983.

VITA

Captain Allan S. Netzer was born on 30 August 1957 in Hawthorne, California. He graduated from high school in 1975 in Titusville, Florida. In December 1980, he graduated from Auburn University with a Bachelor of Aerospace Engineering degree, and was Commissioned through the Air Force Reserve Officer Training Corps. From 1982 to 1984, he served as a systems engineer in the Aeronautical Systems Division, Wright Patterson AFB. Captain Netzer entered the Air Force Institute of Technology in May 1984 to pursue a Masters of Aeronautical Engineering degree.

Permanent Address:
1145 Sharon Drive
Titusville, FL 32796

unclassified

SECURITY CLASSIFICATION OF THIS PAGE

AD A163 830

REPORT DOCUMENTATION PAGE

REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/71E/ENG/85D-2			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION School of Engineering Air Force Institute of Tech		6b. OFFICE SYMBOL (If applicable) AFIT/EN		7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State and ZIP Code) Wright-Patterson AFB, Ohio 45433 Air Force Institute of Technology				7b. ADDRESS (City, State and ZIP Code)	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Air Force Weapons Lab		8b. OFFICE SYMBOL (If applicable) AFWL/ARIA		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State and ZIP Code) Kirtland AFB, NM 87117				10. SOURCE OF FUNDING NOS.	
				PROGRAM ELEMENT NO.	PROJECT NO.
11. TITLE (Include Security Classification) See Box 19					
12. PERSONAL AUTHOR(S) William G. Netzer, BAE, Capt., USAF					
13a. TYPE OF REPORT Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Yr., Mo., Day) 1985, December	
15. PAGE COUNT 315					
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Adaptive Estimation, Image Tracking, Kalman Filter, Target Tracking, Multiple Model Adaptive Filter, Maximum a Posteriori estimator, Forward looking Infrared (FLIR)		
FIELD	GROUP	SUB GR			
19. ABSTRACT (Continue on reverse if necessary and identify by block number) IDENTIFICATION OF BAYESIAN MULTIPLE MODEL ADAPTIVE FILTERING FOR TRACKING AIRBORNE TARGETS Thesis Advisor: Dr. Peter G. Maybeck					
20. DISTRIBUTION AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED UNLIMITED <input type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Peter G. Maybeck Professor of Electrical Engineering			22b. TELEPHONE NUMBER (Include Area Code) 516-255-6450		22c. OFFICE SYMBOL AFIT/ENG

Approved for public release: 1AW AFR 190-17.
 L. W. WOLVER
 16 JAN 86
 Dean for Research and Professional Development
 Air Force Institute of Technology (AFIT)
 Wright-Patterson AFB OH 45433

Abstract

Previous studies at the Air Force Institute of Technology have led to the development of a multiple model adaptive filter (MMAF) tracking algorithm which provides significant improvements in tracker performance against highly-dynamic airborne targets, over the currently used correlation trackers. A forward looking infra-red (FLIR) sensor is used to provide a target shape function to the tracking algorithm in the form of an 8 x 8 array of intensities projected onto a field of view (FOV). This target image measurement is correlated with an estimate of the target image, a template, to produce linear offset pseudo-measurements from the center of the FOV, which are provided as measurements to a bank of linear Kalman filters, in the multiple model adaptive filtering (MMAF) structure. The output of the MMAF provides the state estimates used in pointing the FLIR sensor, and generating the new target image estimate. This study investigates the characteristics of this algorithm in order to evaluate its performance against various target scenarios.

END

FILMED

3-86

DTIC